



Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL DE FI DE CARRERA

TÍTOL DEL TFC: Gas Turbine Design & Analysis Tool: Heat Transfer Components

TITULACIÓ: Enginyeria Tècnica Aeronàutica, especialitat Aeronavegació

AUTOR: Jose Vigo Medina

DIRECTOR: Fernando Pablo Mellibovsky Elstein

DATA: 16 de juliol de 2012

Títol: Gas Turbine Design & AnalysisTool: Heat Transfer Components

Autor: Jose Vigo Medina

Director: Fernando Pablo Mellibovsky Elstein

Data: 16 de juliol de 2012

Resum

Este trabajo tiene como objetivo la modelización de diferentes componentes que son parte o están relacionados con la transferencia de calor (aporte, intercambio o sustracción) de una turbina de gas. Todos estos módulos se han ido desarrollando escalonadamente desde el caso más ideal hasta un cierto grado de complejidad para su posterior programación y puesta en práctica.

El resultado final consta de una serie de módulos programados y testeados en C++ para que puedan simular de forma adecuada los diferentes componentes que representan. En concreto, encontramos los módulos de la cámara de combustión, el sistema de postcombustión y el intercambiador de calor. Por otra parte, se han diseñado otra serie de componentes no relacionados con la transferencia de calor, pero sí con las de una turbina de gas. Estas son una atmosfera estándar, la variación de las características del fluido a lo largo de su paso por el motor y el cálculo de los parámetros finales del motor.

Además, todo este trabajo forma parte de un proyecto mayor en el que se ha tratado de crear un programa capaz de ensamblar cualquier tipo de sistema propulsor y de forma que el usuario pueda diseñar su propio motor mediante la elección de los componentes que él estime oportuno.

Se espera que este trabajo sea el primer paso hacia la creación de un software maduro para el cálculo de diferentes parámetros en cualquier tipo de aplicación de turbina de gas. Aun así, son muchos los obstáculos que se tendrán que superar para conseguir llevar a cabo esta ardua, aunque gratificante tarea.

Title: Gas Turbine Design & Analysis Tool: Heat Transfer Components

Author: Jose Vigo Medina

Director: Fernando Pablo Mellibovsky Elstein

Date: July, 16th 2012

Overview

This work aims the modeling of different components that are part of or related to the transfer of heat (contribution, exchange or removal) from a gas turbine. All these modules have been developed in steps from the most ideal case to a certain degree of complexity for subsequent planning and implementation.

The final result is a series of modules programmed and tested in C++ so that they can adequately simulate their respective components. Specifically, we find the modules of a combustion chamber, an afterburner and a heat exchanger. Moreover, we have designed a series of other components not related to heat transfer, but with those of a gas turbine. These are a standard atmosphere, the variation of the characteristics of the fluid along its passage through the engine and the final calculating of the parameters of the engine.

Moreover, all this work is part of a larger project which has sought to create a program capable of assembling any type of propulsion system and so the user can design their own engine by choosing the components that it considers appropriate.

It is hoped that this work becomes the first step towards the creation of a mature software for the calculation of various parameters on any type of application of a gas turbine. Still, many obstacles need to be overcome before this arduous, but rewarding, task is completed.

Me gustaría dedicar este trabajo a todas aquellas personas que han hecho que sea posible y que me han mostrado su apoyo y comprensión. En concreto a mi tutor Fernando, por su atención e interés constante en este trabajo, a mis dos compañeros de modelización de motores de turbina, Ferran y David, a toda mi familia por su apoyo incondicional y en especial a mi novia Sara por su paciencia y soporte continuo a lo largo de todo este duro recorrido.

ÍNDIX

CAPÍTULO 1. INTRODUCCIÓN	1
1.1. Fundamentos teóricos de las turbinas de gas	1
1.1.1. Introducción	1
1.1.2. Ciclo de Brayton	2
1.2. Sistemas propulsivos	4
1.2.1. Tipos de sistemas propulsivos	4
1.2.2. Turborreactores	4
1.2.3. Turboeje.....	6
1.2.4. Turbohélice	6
1.2.5. Estatorreactor	7
1.2.6. Pulsorreactor	9
 CAPÍTULO 2. COMPONENTES DE TRANSFERENCIA DE CALOR.....	 11
2.1. Cámara de combustión.....	11
2.1.1. Introducción sobre cámaras de combustión.....	11
2.1.2. Grado de modelización 0.....	12
2.1.3. Grado de modelización 1.....	13
2.1.4. Grado de modelización 2.....	14
2.1.5. Grado de modelización 3.....	17
2.1.6. Programación del módulo.....	17
2.2. Postcombustión.....	19
2.2.1. Introducción a los sistemas de postcombustión.....	19
2.2.2. Grado de modelización 0.....	20
2.2.3. Grado de modelización 1.....	20
2.2.4. Grado de modelización 2.....	20
2.2.5. Programación del módulo.....	22
2.3. Intercambiador de calor	23
2.3.1. Introducción sobre intercambiadores de calor.	23
2.3.2. Grado de modelización 0.....	24
2.3.3. Grado de modelización 1.....	25
2.3.4. Grado de modelización 2.....	26
2.3.5. Programación del módulo.....	27
 CAPÍTULO 3. OTROS COMPONENTES	 28
3.1. Atmosfera estándar	28
3.1.1. Introducción a atmosfera estándar	28
3.1.2. Grado de modelización 0.....	29
3.1.3. Grado de modelización 1.....	30
3.1.4. Grado de modelización 2.....	30
3.1.5. Programación del módulo.....	31
3.2. Propiedades del fluido	32
3.2.1. Acerca de las propiedades del fluido	32
3.2.2. Grado de modelización 0.....	32
3.2.3. Grado de modelización 1.....	32
3.2.4. Grado de modelización 2.....	32

3.2.5. Programación del módulo.....	33
3.3. Parámetros del motor	34
3.3.1. Introducción a parámetros del motor.....	34
3.3.2. Grado de modelización 0.....	34
CAPÍTULO 4. MÉTODO DE RESOLUCIÓN NEWTON-RAPHSON.....	35
4.1. Métodos de resolución de ecuaciones no lineales.....	35
4.1.1. Base teórica.....	35
4.1.2. Principales métodos de resolución.....	35
4.1.3. Método de Newton-Raphson.....	36
4.2. Programación del método de Newton-Raphson	37
4.2.1. Programación del método para una sola variable.....	37
4.2.2. Programación del método para un sistema multivariable	38
CAPÍTULO 5. RESULTADOS DE LA IMPLEMENTACIÓN DE LOS MÓDULOS.....	41
5.1. Ratio de convergencia del método de Newton.....	41
5.1.1 Resultados del ratio de convergencia para el sistema de postcombustión.	41
5.2. Test de prueba de la cámara de combustión	42
5.2.1. Resultados obtenidos en la cámara de combustión.....	43
5.2.2. Análisis de errores y comparación	45
5.3. Test de prueba del sistema de postcombustión	46
5.3.1. Resultados obtenidos en el sistema de postcombustión	46
5.3.2. Análisis de resultados y comparación	47
5.4. Test de prueba del intercambiador de calor	47
5.4.1. Análisis de resultados obtenidos en el intercambiador	48
5.5. Test de prueba del motor ensamblado tipo 0.....	50
5.6. Test de un estatorreactor	51
CAPÍTULO 6. CONCLUSIONES	53
BIBLIOGRAFÍA	54
ANEXOS	56
A1. Cálculo de la temperatura de transferencia de pared de cámara.....	56
A2. Tabla de valores del coeficiente global de temperatura	58
A3. Gráficas de emisiones	60
A4. Lista de fichas de programación	63
A5. Código de programación (Newton-Raphson).....	83

CAPÍTULO 1. INTRODUCCIÓN

1.1. Fundamentos teóricos de las turbinas de gas

1.1.1. Introducción

A día de hoy la gran mayoría de aeronaves comerciales dedicadas al transporte aéreo utilizan turbinas de gas para realizar sus actividades. Por ello, con el paso de los años, ha surgido la necesidad de crear nuevos diseños y mejoras más significativas en este tipo de motores para reducir el consumo de combustible, reducir emisiones sonoras, aumentar el rango de operación y en general aumentar sus prestaciones.

Es por esto que se ha hecho imprescindible la utilización de software para la simulación de los diferentes componentes que forman estos motores. Actualmente la mayoría de este tipo de software queda limitado a los profesionales que se encargan de diseñarlos.

Nuestro objetivo se basa en proporcionar una herramienta de calidad para la simulación de los diferentes componentes que forman cualquier sistema propulsivo. Además de proporcionar bloques enteros de motores ensamblados, nuestro programa será capaz de simular conjuntos enteros de diferentes tipos de motores, donde el usuario será capaz de escoger los componentes que lo forman. Tenemos que señalar que la intención de este software es que sea totalmente gratuito y libre para el público que esté interesado en este tipo de materias.

En este proyecto en particular se tratarán de implementar y testear, mediante el lenguaje de programación de alto nivel C++ y el compilador proporcionado por “visual studio 2010”, una serie de componentes de una turbina de gas que están relacionados con la transferencia de calor. Concretamente la cámara de combustión, intercambiadores de calor y postcombustión. Todo ello siguiendo un proceso de escalado de modelización que contemplará desde el caso más ideal hasta el caso más complejo. Para entrar en materia de turbinas de gas en esta sección se tratarán de exponer los fundamentos básicos de este tipo de motores.

Podemos definir la turbina de gas como una máquina que transforma directamente la energía del combustible en energía cinética dentro de un ciclo mecánico. Este ciclo está caracterizado por una repetición periódica de una serie de sucesos sin que el fluido inicial retorne a las condiciones iniciales.

Esta máquina basa su movimiento en la tercera ley de Newton. En ella se expone que para toda acción existe una reacción igual en la misma dirección pero en sentido contrario. Esta ley se conoce como Principio de acción y reacción. No obstante, debemos señalar que no todas las turbinas de gas tienen como objetivo el movimiento por reacción. Algunas simplemente son aprovechadas como motor para transmitir una cierta velocidad angular a un eje

y con esta, ser capaces de generar energía eléctrica a través de un generador, transmitir energía mecánica a una caja de accesorios o simplemente transmitir presión neumática.

Debemos señalar que se han empleado unidades del sistema internacional para todas las ecuaciones mencionadas en todo el documento salvo algunos casos concretos señalados.

1.1.2. Ciclo de Brayton

El ciclo de Brayton no es más que una representación de la evolución termodinámica de la masa de aire/gas durante todo el recorrido por el motor a partir de un ciclo cerrado. En este ciclo, se toman las siguientes consideraciones:

- Durante todo el ciclo, el gas que recorre todo el motor se considera un gas ideal.
- Se considera que el fluido de entrada del motor es el mismo que el de la salida.
- Los procesos de compresión y expansión se realizan de forma adiabática, es decir, no existe intercambio de calor térmico con el exterior.
- El proceso de combustión es isobárico, es decir, se realiza a presión constante.

El ciclo de Brayton está integrado por cuatro procesos internamente reversibles:

- 1-2 Compresión isentrópica (en un compresor).
- 2-3 Adición de calor a presión constante.
- 3-4 Expansión isentrópica (en una turbina).
- 4-1 Rechazo de calor a presión constante.

Los diagramas correspondientes a este ciclo ideal T-s (temperatura-entropía) y P-v (Presión-volumen) se muestran en la figura 1.1.

Cuando los cuatro procesos del ciclo de Brayton se ejecutan en dispositivos de flujo estacionario, y los cambios en las energías cinéticas y potenciales son insignificantes, el balance de energía por unidad de masa puede expresarse como:

$$(q_{entrada} - q_{salida}) + (W_{entrada} - W_{salida}) = (h_{salida} - h_{entrada}) \quad (1.1)$$

Donde la cantidad de calor hacia el fluido y desde el fluido de trabajo es:

$$q_{entrada} = h_3 - h_2 = cp(T_3 - T_2) \quad (1.2)$$

$$q_{entrada} = h_4 - h_1 = cp(T_4 - T_1) \quad (1.3)$$

Entonces, la eficiencia térmica pasa a ser:

$$\eta_{ter,Brayton} = \frac{W_{neto}}{q_{entrada}} = 1 - \frac{q_{salida}}{q_{entrada}} \quad (1.4)$$

Y mediante las relaciones isentrópicas:

$$\frac{T_2}{T_1} = \left(\frac{P_2}{P_1}\right)^{\frac{k-1}{k}}, \left(\frac{P_3}{P_4}\right)^{\frac{k-1}{k}} = \left(\frac{T_3}{T_4}\right) \quad (1.5) \quad (1.6)$$

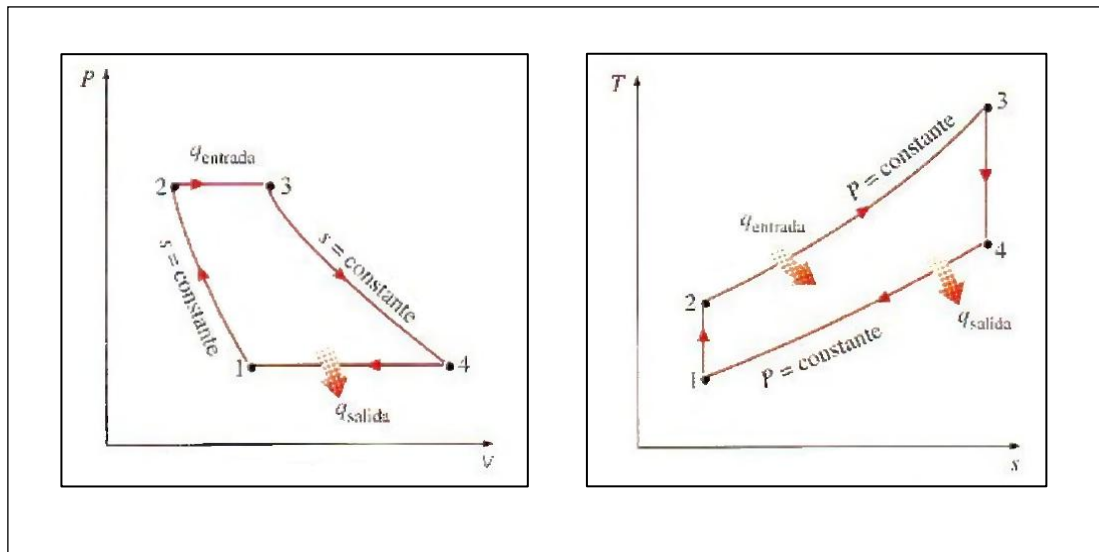


Fig. 1.1 Diagramas T-s y P-v para un ciclo Brayton ideal [9]

La eficiencia térmica y simplificada es:

$$\eta_{ter,Brayton} = 1 - \frac{1}{r_p^{\frac{k-1}{k}}} \quad (1.7)$$

Donde r_p es la relación de presión (P_2/P_1) y k es la relación de calores específicos. [9]

1.2. Sistemas propulsivos

1.2.1. Tipos de sistemas propulsivos

Dentro de este apartado examinaremos los diferentes tipos de sistemas propulsivos, que nuestro programa podría ser capaz de ensamblar (salvo para motores alternativos y motores de cohete). Podemos observar un esquema de estos en la figura 1.2.

Sistemas propulsivos		
Motopropulsores (motores a reacción)	Turborreactores	Flujo único
		Doble flujo
	Estatorreactores	
	Pulsorreactores	
	Motores de cohete	Propulsante sólido
		Propulsante líquido
Grupos motopropulsores	Turbohélices	
	Motor alternativo y hélice	

Fig. 1.2 Tipos de sistemas propulsivos

1.2.2. Turborreactores

Los turborreactores son un tipo de motor a reacción en los que el gas que fluye por el sistema es sometido a una compresión y a una expansión de origen mecánico.

Este tipo de motores se pueden subdividir en dos grupos:

- De flujo único. Estos utilizan todo el flujo de gas absorbido por el difusor para el proceso de admisión, compresión, combustión y posteriormente la expansión en la turbina.
- De doble flujo. Estos dividen el flujo inicial en dos. El flujo primario es aprovechado para el proceso de compresión, combustión y escape. El flujo secundario es acelerado únicamente por la acción de los álabes del “Fan” o de un compresor de baja elevación de presión. Con ello obtenemos una mejora en el rendimiento propulsivo respecto al turborreactor de flujo único.

A continuación se analizará el sistema turborreactor más simple desde el punto de vista mecánico para obtener una visión general de los módulos que se tratarán de programar en este trabajo.

En la figura 1.3 observamos los diferentes componentes del turborreactor básico:

- Difusor de entrada. Tiene como objetivo canalizar el flujo de aire hacia el interior del motor consiguiendo un perfil de velocidad libre de perturbaciones. Todo esto siendo capaz de transformar la mayor parte de energía cinética en presión y temperatura.
- Compresor. Se encarga de aumentar la energía del fluido mediante elevación de la presión a través de la aplicación de un trabajo mecánico. Puede estar formado por varias etapas y suele estar mecánicamente unido por un eje a la turbina, de modo que esta le proporcione la energía necesaria para comprimir el fluido.

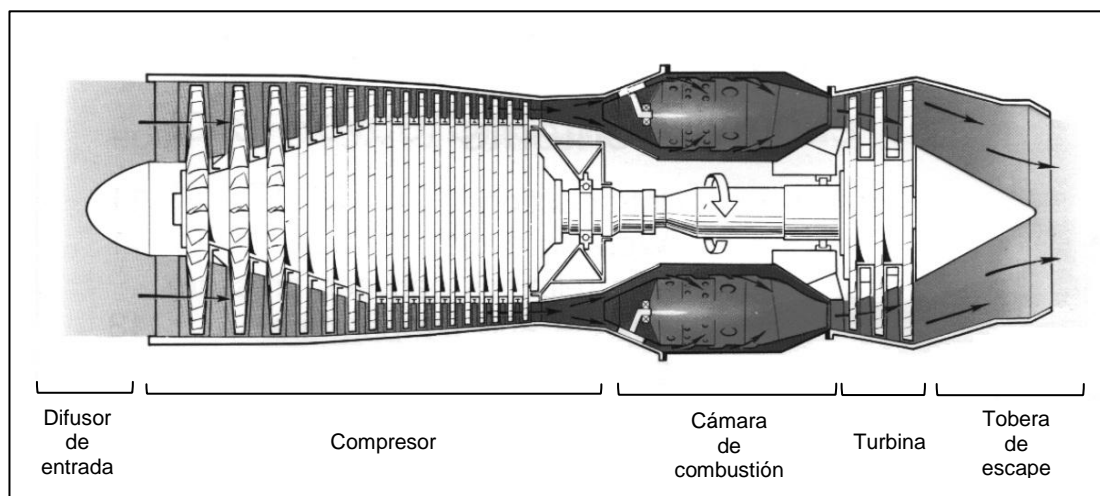


Fig. 1.3 Turborreactor básico de flujo único [6]

- Cámara de combustión. Tiene como objetivo transformar la energía del combustible en energía calorífica. Este proceso, denominado proceso de combustión, se realiza en condiciones ideales a presión y velocidad del fluido constantes.
- Turbina. Es el elemento que se encarga de transformar la energía global del fluido en energía mecánica para mover el eje que conecta con el compresor y accesorios. En la turbina el fluido pierde presión y temperatura manteniendo su velocidad. Hay que señalar que la energía cinética restante de los gases son empleados para producir empuje en el motor.
- Tobera de escape. En este conducto los gases se expanden, es decir, el fluido pierde presión y temperatura para así ganar velocidad y con ello empuje. Además tiene la función de dirigir los gases al exterior de forma que no se produzcan demasiadas perturbaciones para reducir los niveles de ruido. En función del tipo de régimen en el

que la tobera este trabajando, subsónico o supersónico, esta tendrá una geometría diferente.

Para obtener más información sobre los diferentes tipos de configuraciones de este tipo de sistemas propulsivos y sus características mirar las referencias. [5,6,7]

1.2.3. Turboeje

Su funcionamiento es muy similar al del turborreactor con la diferencia que la turbina de estos motores suministra más potencia que la requerida por el compresor. Este exceso queda disponible para el eje de salida de la turbina para proporcionar una velocidad angular al mecanismo al que esté conectado, ya sea un generador, el rotor de un helicóptero o una caja de accesorios. Puede realizarse a través de una caja de engranajes para reducir la velocidad angular o dependiendo de los casos con un sistema de embrague. En la figura 1.4 podemos observar un esquema de un turboeje.

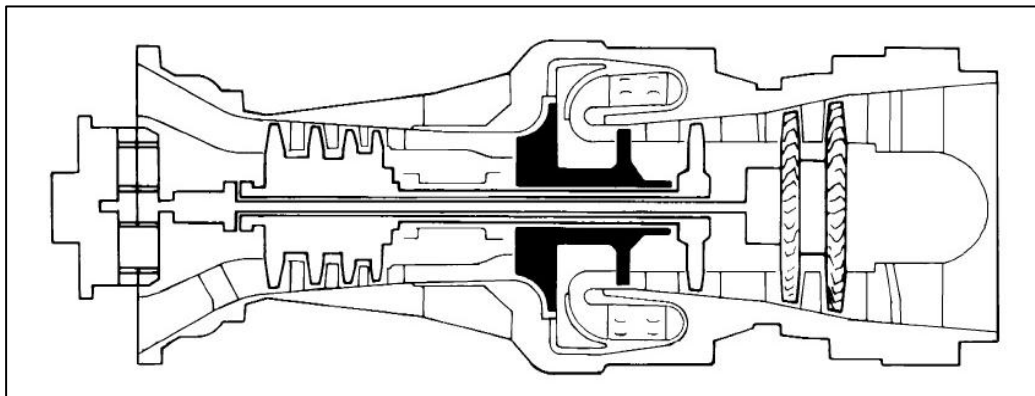


Fig. 1.4 Turboeje con cámara de combustión de flujo reversible [5]

1.2.4. Turbohélice

El turbohélice es un sistema propulsivo muy similar al turborreactor básico. Su ciclo termodinámico es exactamente igual al de un reactor puro. La diferencia más significativa es que gran parte del empuje que proporciona se basa en el arrastre de una hélice por la energía cinética de los gases de expansión de la turbina. A diferencia del turborreactor, cuyo empuje lo proporcionan directamente la velocidad de los gases procedentes de la turbina.

Los motores de turbohélice pueden adoptar diversas configuraciones en función de las necesidades, pero su funcionamiento sigue siendo muy parecido al de los turborreactores de doble flujo, en el que el flujo secundario es producido por la hélice, y el flujo primario por el propio turborreactor.

En los turbohélices modernos, el conjunto reductor y la hélice son arrastradas por una turbina independiente denominada “turbina libre”, ya que en caso de que las hélices sufrieran algún tipo de rotura o impacto, los daños al motor sean mínimos.

Este tipo de motores suelen estar limitados por la formación de ondas de choque en las puntas de las hélices, por lo que su régimen de funcionamiento no suele superar las 1500-2000 revoluciones por minuto dependiendo de las hélices. La razón por la que se siguen empleando este tipo de motores hoy en día es por su bajo consumo de combustible y su facilidad de mantenimiento. En la figura 1.5 se puede apreciar el esquema de un motor turbohélice de compresor de dos etapas y flujo axial. [5]

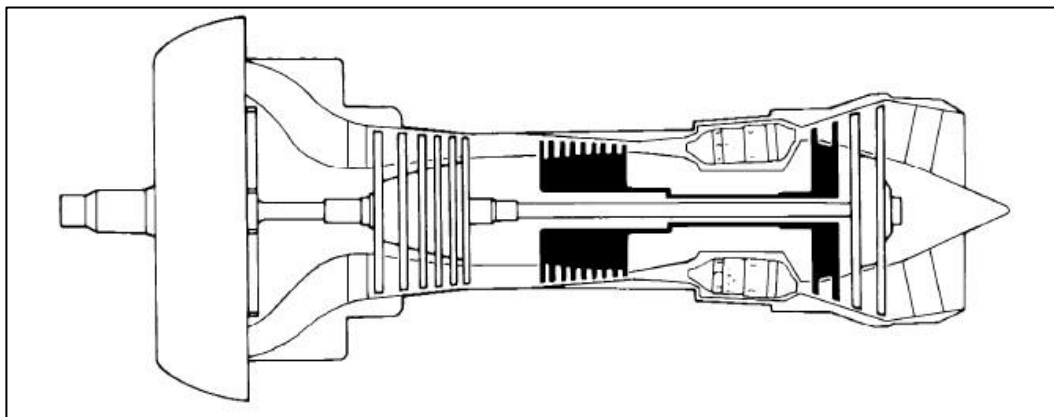


Fig. 1.5 Turbohélíce de compresor de dos etapas y flujo axial [5]

1.2.5. Estatorreactor

El estatorreactor es un motor a reacción que carece de los elementos principales de las turbo máquinas, es decir, no posee compresores ni turbinas. Además no posee piezas mecánicas móviles, a excepción de la bomba de combustible. No obstante, como en cualquier motor a reacción es necesario que se produzca un cierto grado de compresión y expansión de forma que se pueda realizar un proceso de combustión de forma satisfactoria. Al no contar con compresores, se hace uso de la compresión por alta presión dinámica obtenida de la alta velocidad a la que es necesario situar el estatorreactor para su funcionamiento. Sus componentes principales vienen señalados en la figura 1.6.

Su proceso de funcionamiento sigue las siguientes fases:

1. El aire es comprimido por la alta presión dinámica.
2. Se realiza una inyección de combustible atomizado.

3. A través de la bujía se genera una chispa de encendido que sitúa el estatorreactor en un funcionamiento continuo.
4. Se realiza un proceso de combustión en la cámara.
5. Se produce una expansión en la tobera de escape hasta la presión atmosférica para aprovechar el máximo empuje.

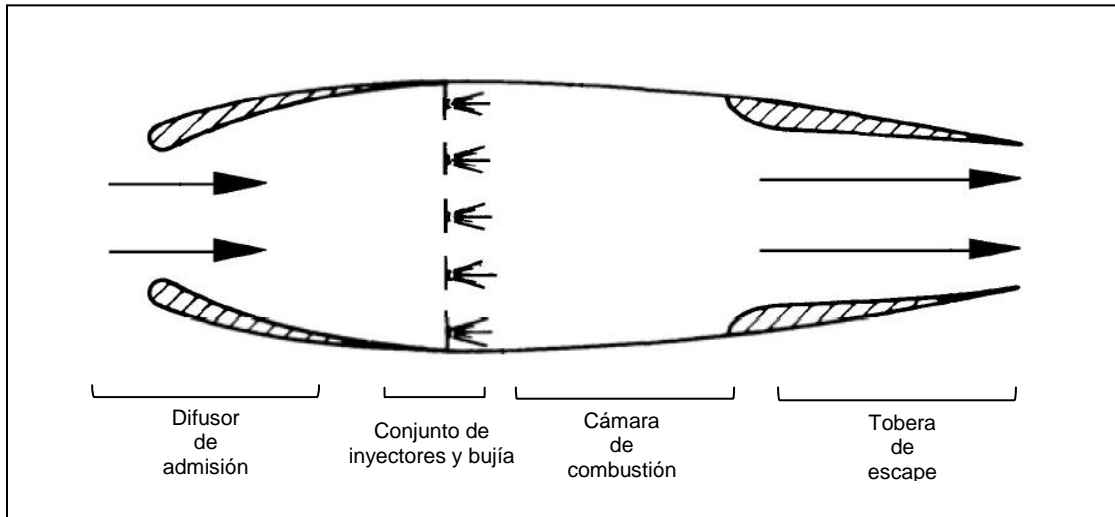


Fig. 1.6 Estatorreactor básico [10]

En función del tipo de régimen sobre el que esté trabajando, el estatorreactor utilizará una configuración del difusor de admisión y de la tobera de escape distinta. Para velocidades subsónicas el difusor tiene forma divergente y la tobera de escape convergente. En cambio, para velocidades supersónicas el difusor de entrada y la tobera de escape suelen ser de forma convergente-divergente.

Esta variación de formas se puede apreciar para velocidades subsónicas y supersónicas en las figuras 1.6 y 1.7 respectivamente.

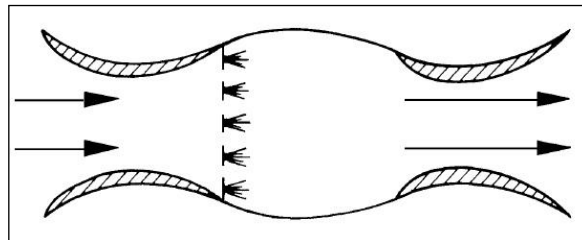


Fig. 1.7 Estatorreactor para velocidades supersónicas [10]

Su principio de funcionamiento es igual a de todos los motores a reacción: La variación de la cantidad de movimiento del aire a la entrada y del gas aire-combustible a la salida.

Las actuaciones de las partes del estatorreactor supersónico (difusor y tobera) poseen ciertas diferencias respecto a las de un estatorreactor subsónico al trabajar a velocidades supersónicas.

- Difusor de admisión para velocidades supersónicas. La elevación de presión necesaria se obtiene al reducir hasta la velocidad del sonido la velocidad del aire en el conducto convergente. En el conducto divergente la presión sigue aumentando y la velocidad alcanza valores subsónicos, imprescindible para que el proceso de combustión tenga éxito. No obstante, este tipo de conducto de admisión tiene el inconveniente de que se puedan producir ondas de choque y superficies de discontinuidad de presiones que puedan ocasionar una disminución del rendimiento.
- Toberas de escape para propulsión supersónica. El tramo convergente aumenta la velocidad hasta la del sonido. Posteriormente en el tramo divergente la velocidad sigue aumentando a expensas de una disminución de presión.

Los mejores rendimientos en los estatorreactores se obtienen a partir de velocidades de 1000Km/h. Este es el principal inconveniente de este tipo de sistemas de propulsión, para su utilización, es necesario situar el motor a velocidades próximas a las del sonido. Además, posee problemas térmicos debido a las altas velocidades del fluido. Esto es debido a la gran dificultad para refrigerar la cámara de combustión por el rozamiento de la capa límite exterior. Por esta razón las paredes de estas cámaras suelen venir revestidas con materiales cerámicos. [10]

1.2.6. Pulsorreactor

El funcionamiento del pulsorreactor es muy similar al del estatorreactor. La diferencia fundamental es que la combustión se realiza de forma intermitente a impulsos de alta frecuencia por la acción de unas válvulas que obstaculizan la admisión del aire.

La combustión se realiza con las válvulas cerradas y a diferencia del estatorreactor, este es capaz de regular la presión para su funcionamiento. De esta forma es capaz de trabajar a velocidades inferiores. Sus componentes principales pueden apreciarse en la figura 1.8

La combustión en los pulsorreactores, a diferencia de todos los sistemas propulsivos que hemos visto hasta ahora, se realiza a volumen constante, similar al que tiene lugar en los motores alternativos, solo que en este caso la combustión se realiza a muy alta frecuencia. Esta es regulada por las válvulas

de intermitencia de admisión. Durante la apertura de dichas válvulas se produce la fase de carga y cuando esta se cierra se produce la fase de combustión y escape de los gases.

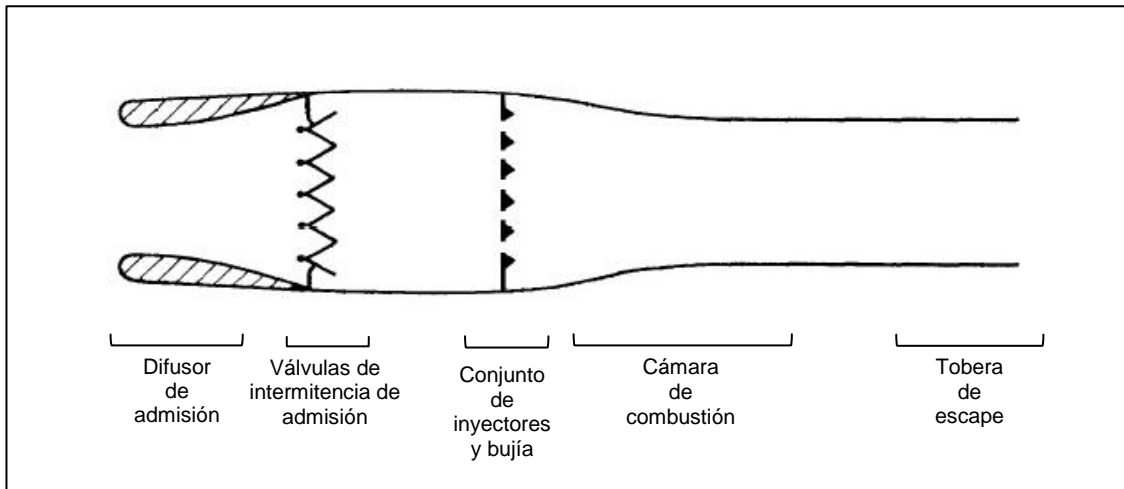


Fig. 1.8 Pulsorreactor básico [10]

El funcionamiento termodinámico del pulsorreactor se corresponde con el ciclo de Lenoir. Inicialmente se produce un incremento de la presión a volumen constante seguida de una expansión hasta verse reducida a la presión inicial de admisión. Puede contemplarse una representación de este ciclo en la figura 1.9.

Pese a que el pulsorreactor pueda producir empuje a velocidades reducidas cuenta con un rendimiento global muy bajo. Además tiene dificultades de funcionamiento mecánico y térmico en las válvulas de intermitencia de admisión por el retardo de funcionamiento de estas. Por esta razón puede verse limitada la velocidad de vuelo. Como consecuencia de todos estos inconvenientes en la actualidad carece de interés aeronáutico. [10]

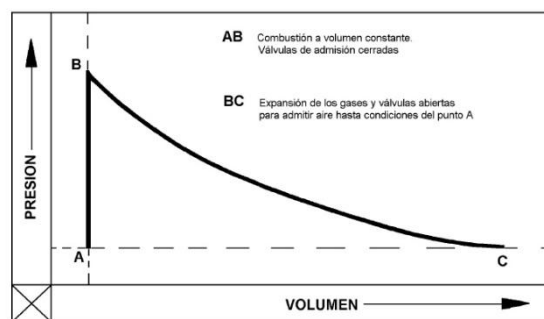


Fig. 1.9 Representación del ciclo de Lenoir

CAPÍTULO 2. COMPONENTES DE TRANSFERENCIA DE CALOR

2.1. Cámara de combustión

2.1.1. Introducción sobre cámaras de combustión

Las cámaras de combustión son dispositivos que se encargan de introducir una cierta cantidad de energía calorífica sobre un fluido. Normalmente, en las cámaras de combustión de una turbina de gas, este proceso suele realizarse a presión constante y la adición de calor se ve reflejada en el incremento de temperatura del fluido.

A día de hoy existen tres tipos de cámaras de combustión para turbinas de gas:

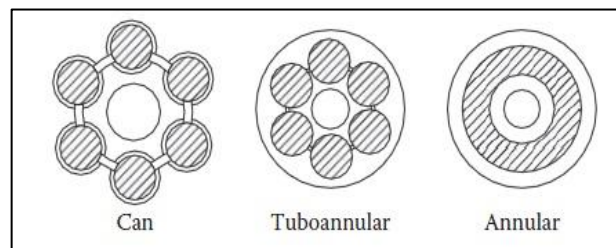


Fig. 2.1 Tipos de cámaras de combustión [1]

-Cámaras tubulares o “can”. Este tipo de cámaras se componen por un revestimiento cilíndrico montado concéntricamente en una carcasa cilíndrica tal y como muestra la figura 2.1. La principal ventaja de este tipo de sistemas es su reducido coste de fabricación. Sin embargo, su longitud y peso excesivos ha hecho que este tipo de cámaras no se utilice en aeronaves, sino más bien en unidades industriales.

-Cámaras tubo-anulares. El diseño de este tipo de cámaras se basa en un grupo de forros tubulares, generalmente de 6 a 10, dispuesto en el interior de una carcasa anular tal como ilustra la figura 2.1. Este concepto de cámaras trata de combinar la compactación de la cámara anular con la resistencia mecánica de la cámara tubular. Su principal inconveniente deriva de la necesidad de tener interconexiones de llama (tubos de fuego cruzado).

-Cámaras anulares. Este tipo de cámaras poseen un revestimiento anular montado concéntricamente en una carcasa anular tal y como muestra la figura 2.1. Esta es sin duda, una de las cámaras ideales en cuanto a limpieza aerodinámica. El resultado de este diseño deriva en una unidad compacta y unas pérdidas de presión menores que en los otros tipos de cámaras. Su

principal inconveniente viene ocasionado por su excesiva carga de pandeo en el revestimiento exterior.

Aun así, existen un gran número de variaciones dentro de estos tres grupos. Tratan de jugar con el diseño de la geometría, los tipos de inyectores de combustible o incluso la orientación de la propia cámara. Todo ello con tal de adaptarse a las necesidades del tipo de unidad de motor que se necesite. No obstante todas las cámaras tienen en común las zonas en las que se divide el proceso de combustión tal como se ilustra en la figura 2.2. Como observaremos en grados de modelizaciones dos y tres, la diferenciación de estas zonas resulta fundamental para el cálculo del valor simulado de emisiones o la temperatura de transferencia de la pared de cámara.

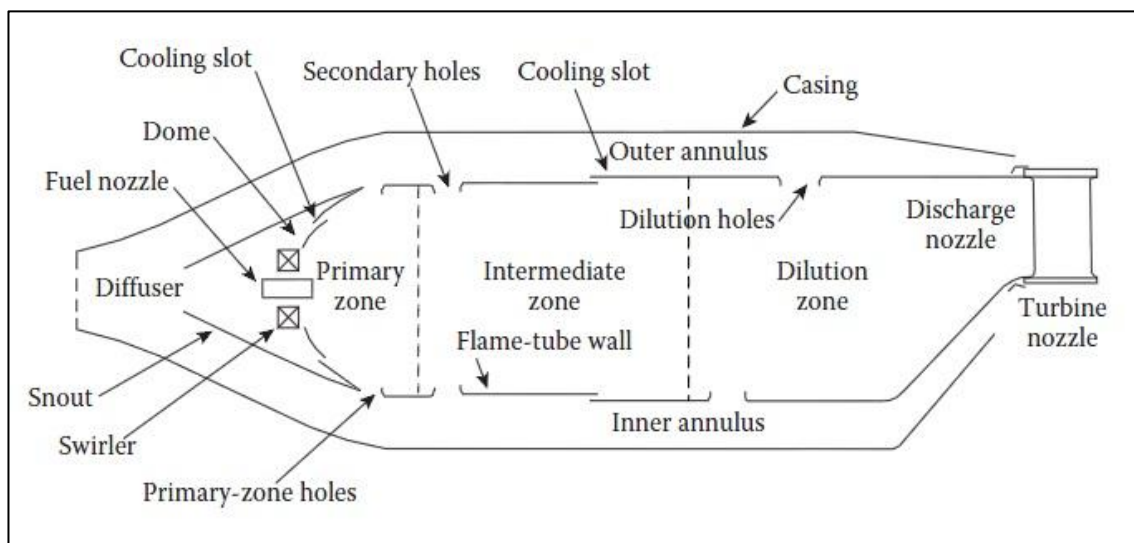


Fig. 2.2 Zonas comunes de una cámara de combustión [1]

Para una correcta modelización de las cámaras de combustión se debe tener en cuenta que entran en juego aspectos de transferencia de calor, transferencia de masa, termodinámica, dinámica de gases y dinámica de fluidos. A continuación se describirán los diferentes grados de modelización de las cámaras.

2.1.2. Grado de modelización 0

La idea para este grado de modelización se basa en tratar de modelar una cámara de combustión ideal con las siguientes suposiciones:

1. La adición de calor se produce a presión constante sin ningún tipo de pérdidas.
2. El proceso de combustión se produce en condiciones ideales.

3. Consideramos el fluido que entra en la cámara como un gas perfecto con un calor específico constante e invariable composición. No obstante, los valores del calor específico pueden variar en función del grado de modelización del módulo de tipo de fluido (sección 3.2).
4. Todo el proceso de combustión se realiza de forma adiabática, es decir, sin pérdidas de calor hacia el exterior.

Para este caso las únicas ecuaciones que intervienen son:

$$\dot{Q}_{in} = (h_{out} - h_{in}) = \dot{m} * cp(T_{out} - T_{in}) \quad (2.1)$$

$$\dot{Q}_{in} = \dot{m}_{fuel} * q_{fuel} \quad (2.2)$$

$$P_{in} = P_{out} \quad (2.3)$$

Donde T es la temperatura, tanto de salida (T_{out}) como de entrada (T_{in}), \dot{Q}_{in} es el calor introducido a nuestra cámara, cp es el calor específico a presión constante del gas, \dot{m} es el flujo de masa de aire de entrada en función del tiempo, \dot{m}_{fuel} es el flujo de masa de combustible de entrada en la cámara en función del tiempo, q_{fuel} es el poder calorífico del combustible y P representa la presión en kPa.

2.1.3. Grado de modelización 1

En este grado de modelización se pretende simular las condiciones de una cámara de combustión ideales, pero añadiendo constantes para simular las pérdidas e irreversibilidades del proceso de combustión.

Las hipótesis realizadas en el grado de modelización 0 (visto en la sección 2.1.2 de este documento) se conservan en este salvo por los puntos 1 y 2. En su lugar tenemos:

1. La eficiencia de combustión y el momento de pérdida de presión son definidas por constantes con valores típicos de una cámara de combustión.
2. Consideramos un incremento en el flujo másico del fluido debido a la adición de combustible.

Las ecuaciones empleadas para la modelización se corresponden con la expresión 2.1 (donde $\dot{m} = \dot{m}_{fuel}$) y una pequeña variación de la 2.2 en la que incluimos el factor de eficiencia de combustión η_c cuyo valor es aproximado a 0.95:

$$\dot{Q}_{in} = \eta_c * \dot{m}_{fuel} * q_{fuel} \quad (2.4)$$

$$\eta_d = \frac{P_{in} - P_{out}}{P_{in}} = 0.06 \quad (2.5)$$

$$\dot{m}_{out} = \dot{m}_{in} + \dot{m}_{fuel} \quad (2.6)$$

Consideramos unas pérdidas de presión η_d del 6% y mediante la expresión 2.6 consideramos el incremento del flujo másico por la adición de combustible. No obstante, el usuario será capaz de modificar estos valores para la simulación.

2.1.4. Grado de modelización 2

En este grado de modelización se trata de averiguar esas constantes que definen las pérdidas e irreversibilidades a partir de expresiones empíricas. Éstas se calculan a partir de parámetros referentes a las características de la cámara de combustión. Además en este módulo se incluye una serie de métodos extraídos de la referencia [1] para calcular los niveles de emisiones de cualquier cámara de combustión.

Dentro de este grado de modelización tenemos las siguientes hipótesis:

1. La adición de calor es estimada en función de la relación de combustible/aire, la presión de entrada y la temperatura de entrada a partir de las curvas de incrementos de temperatura para mezclas de keroseno-aire.
2. La eficiencia de combustión es estimada en función de la relación de combustible/aire.
3. El momento de pérdida de presión es definido por constantes con valores típicos dependiendo del tipo de cámara de combustión.

Las ecuaciones empleadas para la modelización se corresponden con la 2.1, 2.3 y la 2.6 de los casos anteriores. El momento de pérdida de presión η_d para cámaras tubulares es considerado con un 7%, mientras que para cámaras tubo-anulares y anulares es considerado de un 6%. Estos valores están extraídos de la referencia [1] y están deducidos a partir de diferentes ensayos con diferentes tipos de cámaras. Además de dichas ecuaciones tenemos las siguientes para este grado de modelización:

$$T_{out} = T_{in} + \Delta T_{comb}(f, T_{in}, P_{in}) \quad (2.7)$$

$$\eta_c = \frac{f_{theoretical}}{f_{actual}} \quad (2.8)$$

Donde f_{actual} representa la relación de combustible/aire actual en la cámara y $f_{theoretical}$ la relación de combustible/aire para una combustión estequiométrica.

La variable ΔT_{comb} es deducida a partir de las curvas de incrementos de temperatura para mezclas de keroseno-aire. Concretamente la que muestra la figura 2.3.

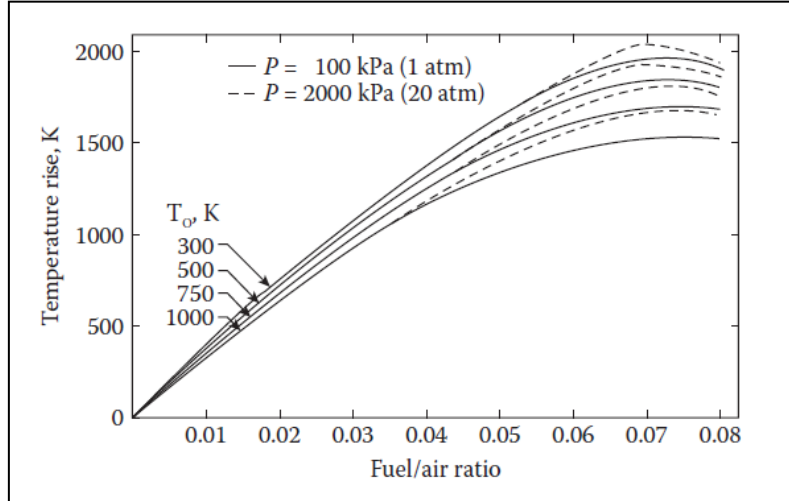


Fig. 2.3 Curvas de incremento de temperatura en mezclas de keroseno-aire [1]

En este módulo se calculará una estimación de las emisiones de NO_x , CO y UHC. A continuación se mostrará el procedimiento que se ha llevado a cabo para realizar este cálculo.

Para la estimación de emisiones de los óxidos de nitrógeno (NO_x) de la cámara se utilizará la ecuación de Lefebvre extraída de la referencia [1]:

$$NO_x = \frac{9 \cdot 10^{-8} P_{in}^{1.25} V_c e^{0.01 T_{st}}}{\dot{m}_{in} T_{pz}} \quad (2.9)$$

Donde V_c es el volumen de combustión, T_{pz} la temperatura en la zona primaria, T_{st} la temperatura estequiométrica de llama, \dot{m}_{in} el flujo de masa de aire de entrada en función del tiempo y P_{in} la presión de entrada en la cámara.

Para el cálculo de algunos de estos parámetros, referentes al diseño de la cámara de combustión, se emplean las siguientes expresiones:

$$V_c (tubular, tuboanular) = n_{t,c} * L * A_{ref} \quad (2.10)$$

$$V_c (anular) = L * (A_{carter exterior} - A_{carter interior}) \quad (2.11)$$

$$T_{pz} = T_{in} + \Delta T_{comb}(0.85 * f_{actual}, P_{in}, T_{in}) - 55 \quad (2.12)$$

$$T_{st} = T_{in} + \Delta T_{comb}(f_{theoretical}, P_{in}, T_{in}) \quad (2.13)$$

Donde $n_{t,c}$ representa el número de cilindros de las cámaras tubulares o tubo-anulares, L la longitud de la cámara, A_{ref} el área de referencia de un “can”, \dot{m}_{in} el flujo de masa de aire de entrada en función del tiempo, P_{in} la presión de entrada en la cámara, $A_{carter\ exterior}$ el área del cárter exterior y $A_{carter\ interior}$ el área del cárter interior.

A través de la ecuación 2.12 se calcula la temperatura en la zona primaria, en la cual utilizaremos como aproximación un 85% de eficiencia respecto a la eficiencia de combustión de la cámara. Además le restamos 55 K para compensar el calor perdido durante la evaporación del combustible no quemado.

Para el cálculo de la temperatura de llama estequiométrica utilizaremos las mismas curvas de incrementos de temperatura para mezclas de keroseno-aire. Con la diferencia de que se considerará una eficiencia de combustión del 100% y por lo tanto se calculara este incremento con la relación de combustible/aire estequiométrica, $f_{theoretical}$. Todo ello se ve reflejado en la ecuación 2.13.

Para la estimación de emisiones de los monóxidos de carbono (CO) se utilizará la aproximación de Lefebvre extraída de la referencia [1]:

$$CO = \frac{86\dot{m}_{in}T_{pz}*e^{(-0.00345T_{pz})}}{(V_c-V_e)\left(\frac{\Delta P}{P}\right)^{0.5}*P_{in}^{1.5}} \quad (2.14)$$

Donde V_c es el volumen de combustión, V_e el volumen ocupado en la evaporación, T_{pz} la temperatura en la zona primaria, \dot{m}_{in} el flujo de masa de aire de entrada en función del tiempo y P_{in} la presión de entrada en la cámara.

Para el cálculo del volumen ocupado en la evaporación se utiliza la siguiente expresión:

$$V_e = 0.55 * \dot{m}_{pz} * D_0^2 / \rho_{pz} \lambda_{eff} \quad (2.15)$$

Donde \dot{m}_{pz} representa el flujo másico en la zona primaria, D_0^2 el diámetro medio de las gotas de evaporación al cuadrado, ρ_{pz} la densidad del fluido en la zona primaria y λ_{eff} la constante de evaporación.

Para calcular el flujo de masa en la zona primaria, \dot{m}_{pz} estimamos que representa un 26% del flujo másico de entrada. Los valores del diámetro medio de las gotas y de la densidad del fluido en la zona primaria son extraídos de

valores típicos de las cámaras de combustión. En cambio, los valores de la constante de evaporación varían en función de la temperatura en la zona primaria y por ello se han debido realizar diferentes casos para su estimación.

Para la estimación de los hidrocarburos no quemados (UHC) se ha utilizado una expresión que la relaciona con las emisiones de CO y la eficiencia de combustión extraída de la referencia [1].

$$UHC = \frac{(1-\eta_c)}{10^{-3}} - 0.232 * CO \quad (2.16)$$

2.1.5. Grado de modelización 3

La idea para este grado de modelización consiste en realizar el mismo tipo de modelización del grado 2, añadiendo una serie de ecuaciones de transferencia de calor para averiguar la temperatura de la pared de la cámara de combustión. Estas fórmulas vienen expresadas en el anexo A1 y han sido extraídas de la referencia [1].

Esta estimación se basa en el cálculo por separado de la radiación interna procedente de los gases no luminosos y los gases luminosos (R_1), la radiación externa (R_2), la convección interna (C_1) y la convección externa (C_2). Esta serie de cálculos sigue el esquema de la figura 2.4.

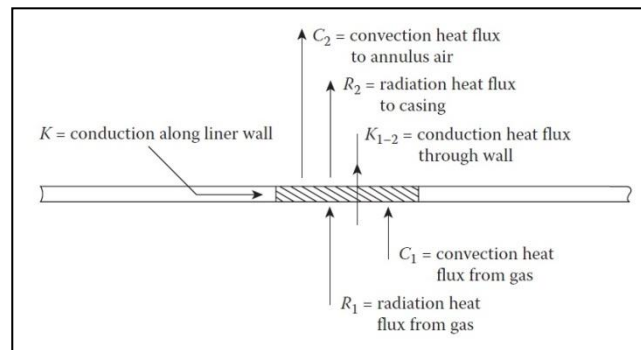


Fig. 2.4 Esquema de los diferentes factores que intervienen en la transferencia de calor de la pared de la cámara de combustión [1]

2.1.6. Programación del módulo

Durante la programación de este módulo ha sido necesaria expresar en forma de ecuaciones las curvas de incrementos de temperatura para mezclas de keroseno-aire. En la figura 2.5 podemos observar una representación de la aproximación realizada mediante una ecuación por partes.

La ecuación por partes de la modelización de las curvas se corresponde con:

$$\Delta T(f) = \begin{cases} 33333.33 * f & 0 \leq f < 0.03 \\ 25000 * f + 250 & 0.03 \leq f < 0.05 \\ 12500 * f + 875 & 0.05 \leq f < 0.07 \\ 1750 & 0.07 \leq f \leq 0.08 \end{cases} \quad (2.17)$$

Para la inserción de los cambios producidos por las variaciones de temperatura de entrada añadimos las siguientes condiciones a la ecuación 2.17:

$$\text{Si } (0.03 \leq f < 0.07) \rightarrow \Delta T = \Delta T(f) + (f - 0.03) * (600 - T_{in}) * 16.25 \quad (2.18)$$

$$\text{Si } (0.07 \leq f < 0.08) \rightarrow \Delta T = \Delta T(f) + 0.02 * (600 - T_{in}) * 30 \quad (2.19)$$

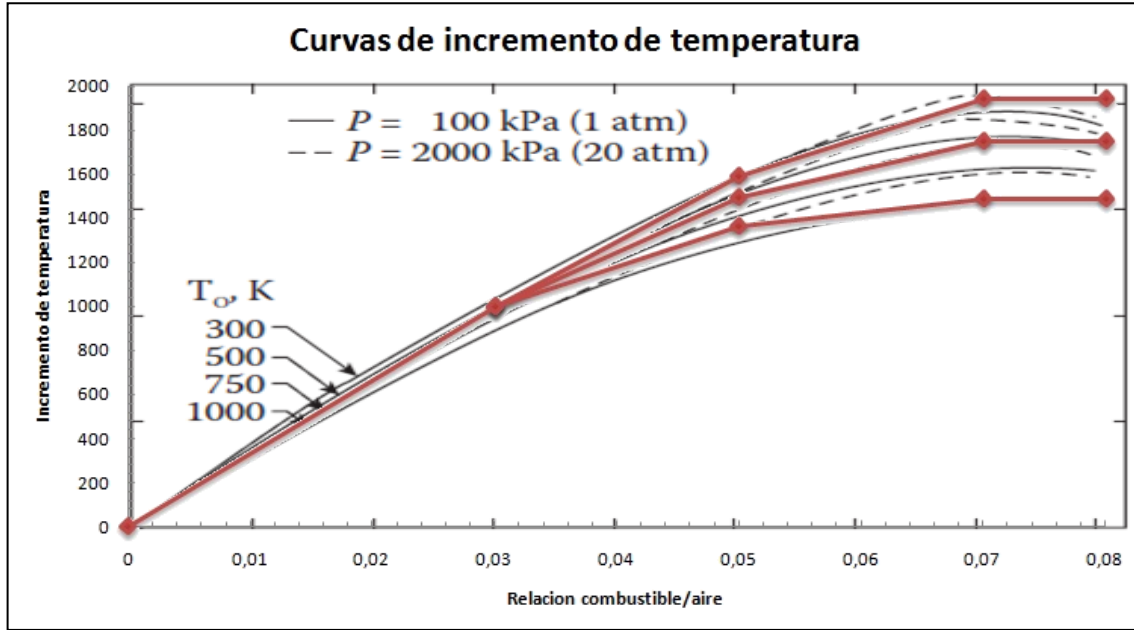


Fig. 2.5 Modelización de las curvas de incremento de temperatura. Las líneas rojas se refieren a la aproximación realizada a través de la ecuación por partes 2.17. Las líneas negras se refieren a la curva real de incremento de temperatura

Para la inserción de los cambios producidos por las variaciones de presión de entrada añadimos la siguiente condición a las ecuaciones 2.17, 2.18 y 2.19:

$$\text{Si } (f > 0.06) \rightarrow \Delta T = \Delta T(f) + (P_{in} - 100 * 10^3) * \frac{19}{190} \quad (2.20)$$

El error final entre las curvas y la aproximación realizada no resulta mayor de 75 K. En el capítulo 5, referido a los resultados de la implementación de los módulos, se estudiará cual es el comportamiento de nuestro sistema respecto a este posible error de aproximación.

2.2. Postcombustión

2.2.1. Introducción a los sistemas de postcombustión.

La postcombustión es un tipo de sistema que proporciona un empuje extra al sistema motor. Este tipo de mecanismos aprovechan el hecho de que los gases de salida de la turbina contienen oxígeno suficiente para realizar una segunda combustión, debido a que solo una parte del aire de entrada al motor es aprovechado para realizar la combustión en las cámaras. La ausencia del posible estrés que tendrían los álabes a estas temperaturas hace posible que se puedan alcanzar temperaturas superiores a las que se dan en la salida de las cámaras de combustión.

El principal inconveniente de este sistema es el incremento del SFC (Specific Fuel Consumption). Por esta razón, la postcombustión acostumbra a usarse durante cortos periodos de tiempo. Algunos turbofans militares suelen incorporar postcombustión para el despegue en espacios reducidos y maniobras de combate.

En la postcombustión existen pérdidas por los cambios de momento resultante de la adición de calor. Estas pérdidas, como se verá reflejado en el punto 2.2.4, se pueden determinar mediante la utilización de funciones de Rayleigh. Para el cálculo de estas pérdidas se deben tener en cuenta las ondas de choque térmicas. Estas, hacen necesaria la inserción de un difusor entre la turbina y la cámara de postcombustión, para reducir el número de Mach a 0.25-0.3 antes de introducir el carburante. Incluso cuando no está en uso se pueden producir pérdidas de presión debido a la presencia de los quemadores. Por este motivo es recomendable la utilización de una tobera convergente de área variable para mejorar las actuaciones de funcionamiento a bajo régimen. Todo el sistema de postcombustión se muestra en la figura 2.6.

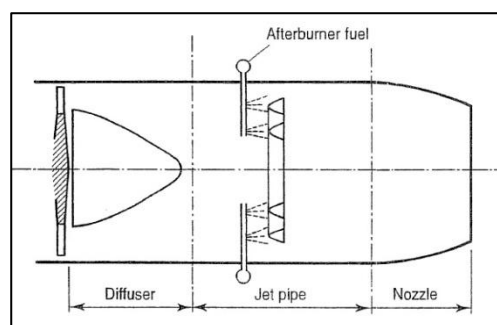


Fig. 2.6 Sistema de postcombustión [2]

2.2.2. Grado de modelización 0

En este grado de modelización se trata de recrear un sistema de post-combustión en condiciones ideales. Por ello se suponen las mismas hipótesis que en la cámara de combustión de grado 0. Esto es debido al hecho de que en condiciones ideales un sistema de postcombustión posee las mismas características y funciones que una cámara de combustión, entregar toda la energía calorífica del combustible al fluido. Todo ello está visto en el punto 2.1.2 de este documento.

2.2.3. Grado de modelización 1

En este grado, el sistema suele ser muy similar al de la cámara de combustión de tipo 1 visto en la sección 2.1.3. Con la excepción, que los valores típicos de la eficiencia de combustión η_c y el momento de pérdida de presión η_d están alrededor del 40% y el 6% respectivamente, característico en las cámaras de postcombustión. Además se siguen utilizando las mismas ecuaciones que las vistas en la sección 2.1.3, concretamente la 2.1, 2.3, 2.4, 2.5 y 2.6.

2.2.4. Grado de modelización 2

En este grado de modelización se ha tratado de averiguar las constantes que definen las pérdidas e irreversibilidades. Para ello se han adoptado las mismas hipótesis que en el grado de modelización 2 de la cámara de combustión (punto 2.1.4 de este documento). Concretamente, en las hipótesis referidas a la estimación de la adición de calor en función de la relación combustible/aire, la temperatura de entrada y la presión de entrada. Y del cálculo de la eficiencia de combustión en función de la relación combustible/aire estequiométrica/actual.

No obstante, para el cálculo de las pérdidas de presión se han utilizado las ecuaciones relevantes de un fluido de Rayleigh para una serie de valores de Mach y la relación de temperaturas de la cámara de postcombustión. Estas ecuaciones se basan en la siguiente hipótesis:

1. Se considera un fluido sin fricción en un conducto de sección constante con una transferencia de calor.

Las ecuaciones utilizadas para la implementación en los módulos para el cálculo del número de Mach y de la pérdida de presión son las siguientes:

$$\frac{P_{out}}{P_{in}} = \frac{P_{out}}{P^*} * \frac{P^*}{P_{in}} \quad (2.21)$$

$$\frac{P_{in}}{P^*} = \left[\frac{1+\gamma}{1+\gamma M_{in}^2} \right] \left[\frac{2+(\gamma-1)M_{in}^2}{\gamma+1} \right]^{\frac{\gamma}{\gamma-1}} \quad (2.22)$$

$$\frac{P_{out}}{P^*} = \left[\frac{1+\gamma}{1+\gamma M_{out}^2} \right] \left[\frac{2+(\gamma-1)M_{out}^2}{\gamma+1} \right]^{\frac{\gamma}{\gamma-1}} \quad (2.23)$$

$$\frac{T_{in}}{T^*} = \frac{(1+\gamma)M^2}{(1+\gamma M^2)^2} [2 + (\gamma - 1)M^2] \quad (2.24)$$

$$\frac{T_{out}}{T^*} = \frac{(1+\gamma)M_{out}^2}{(1+\gamma M_{out}^2)^2} [2 + (\gamma - 1)M_{out}^2] \quad (2.25)$$

$$\frac{T_{out}}{T^*} = \left(\frac{T_{out}}{T_{in}} \right) \left(\frac{T_{in}}{T^*} \right) \quad (2.26)$$

Donde P^* representa la presión de referencia a Mach igual a 1, T^* la temperatura de referencia a Mach igual a 1, P la presión, T la temperatura, M el número de Mach y γ la relación de calores específicos.

Debemos hacer especial hincapié en que todos los valores de presiones y temperaturas son de “stagnation”, es decir, derivan de la cantidad de energía cinética que lleve el fluido. Dado que estamos trabajando con temperaturas y presiones totales todo el tiempo, no tendremos ningún problema en este aspecto.

Para el cálculo de los valores de Mach y pérdida de presión debemos seguir los siguientes pasos:

- Para un Mach de entrada dado, calcular el valor de $\frac{P_{in}}{P^*}$ y $\frac{T_{in}}{T^*}$ a partir de las expresiones 2.22 y 2.24.
- Evaluamos la relación $\frac{T_{out}}{T^*}$ a partir de la expresión 2.26.
- Averiguamos el valor del Mach de salida utilizando la expresión 2.25 a través de la evaluación por medio de $\frac{T_{out}}{T^*}$.
- Resolvemos la relación $\frac{P_{out}}{P^*}$ correspondiente con el valor del número de Mach a la salida, calculado anteriormente, a partir de la expresión 2.23
- Evaluamos finalmente la relación $\frac{P_{out}}{P_{in}}$ a partir de la expresión 2.21.

Podemos observar el comportamiento de esta serie de cálculos en función del Mach de entrada y la relación de temperaturas en la figura 2.7.

Observamos la existencia de una barrera que impide que se produzcan unas pérdidas de presión mayores. Esta barrera es el límite de choque térmico. Este es un aspecto que hay que tener muy en cuenta a la hora de utilizar este tipo de sistemas aumentadores de empuje. Por esta razón y para reducir al máximo

las pérdidas de presión, como ya hemos comentado antes, se suele colocar un difusor para que el fluido entre a una velocidad adecuada de 0.2 - 0.3 Mach.

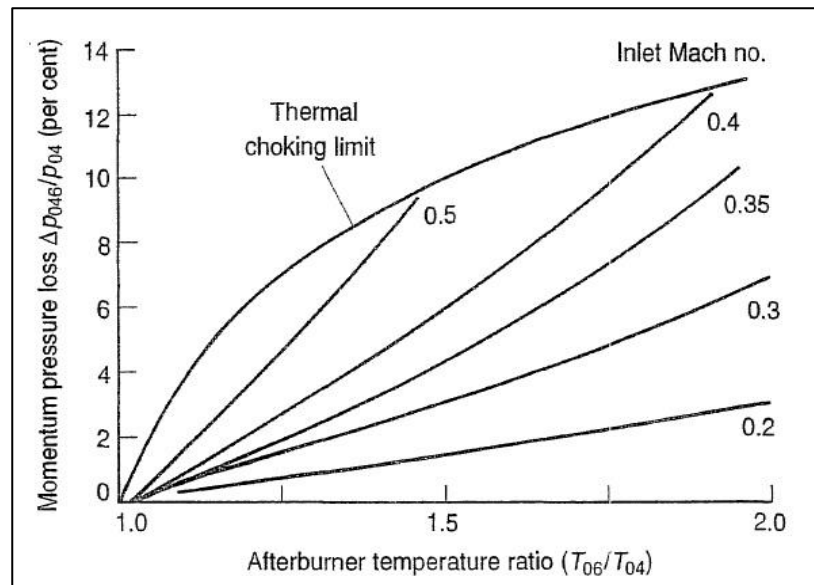


Fig. 2.7 Relación entre las pérdidas de presión y el Mach de entrada y la relación de temperaturas [2]

2.2.5. Programación del módulo

Para la programación de este módulo, esencialmente en el grado de modelización 2, ha sido necesaria la implementación de las curvas de incremento de temperatura para fluidos premezclados de keroseno y aire. Por ello, se ha seguido la misma mecánica que en la cámara de combustión visto en la sección 2.2.5. Además, se han tenido en cuenta ciertos aspectos en la implementación de las ecuaciones que representan un fluido sin fricción en un conducto de sección constante con una transferencia de calor. Al no poder seguir el procedimiento analítico, se han tenido que introducir en las ecuaciones a resolver en el método de Newton.

La razón por la cual no se ha podido seguir el procedimiento analítico es que resulta de vital importancia que el programa no tenga un sentido único, sino que pueda resolverse de forma que se tengan unas incógnitas y unas ecuaciones dadas. Al final, el usuario será capaz de calcular varios factores del módulo para diferentes condiciones iniciales.

Como ya veremos más en profundidad en el capítulo 5, el principal inconveniente de insertar estas ecuaciones en el método es que al tratarse de, en algunos casos, incógnitas elevadas a una potencia 4, debemos escoger de forma muy concreta el diferencial de x y las estimaciones iniciales. El significado y la relevancia de estos parámetros vienen comentados en el capítulo 4.

2.3. Intercambiador de calor

2.3.1. Introducción sobre intercambiadores de calor.

Los intercambiadores de calor son dispositivos donde dos corrientes de fluido intercambian calor sin mezclarse. Su efectividad de transferencia de calor aumenta directamente con el área de superficie de transferencia.

Cuando representamos el valor de la temperatura en función de la longitud del intercambiador se pueden dar dos situaciones típicas. En la primera, la temperatura del fluido frío “t” y la temperatura del fluido caliente “T” varían simultáneamente. En esa situación, el intercambio de calor se describe sin cambio de fase de ninguna de las dos corrientes. Además, las dos corrientes pueden circular en sentidos opuestos o en sentido de corrientes paralelas. En la figura 2.8 y 2.9 podemos observar el esquema y la progresión de las temperaturas para disposiciones a contracorriente y a corrientes paralelas respectivamente. En la segunda situación uno de los dos fluidos experimenta un cambio de fase, es decir, su temperatura permanece constante, mientras que la del otro fluido si varía. Este caso se puede dar para disposiciones a contracorriente. Podemos observar una ilustración de este caso en la figura 2.10. [4]

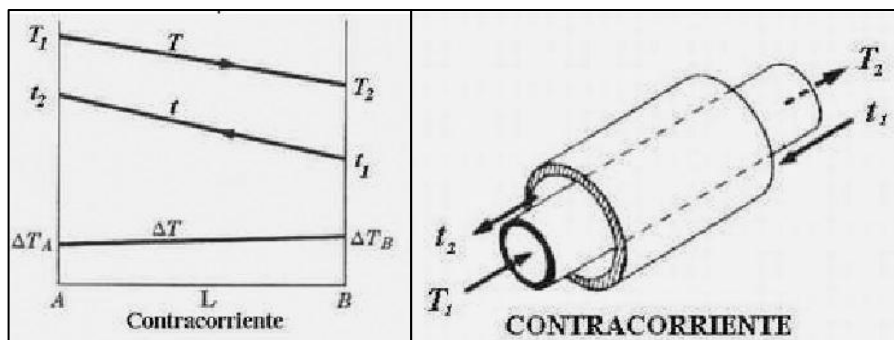


Fig. 2.8 Caso 1. Progresión de las temperaturas y esquema para disposiciones de flujos a contracorriente

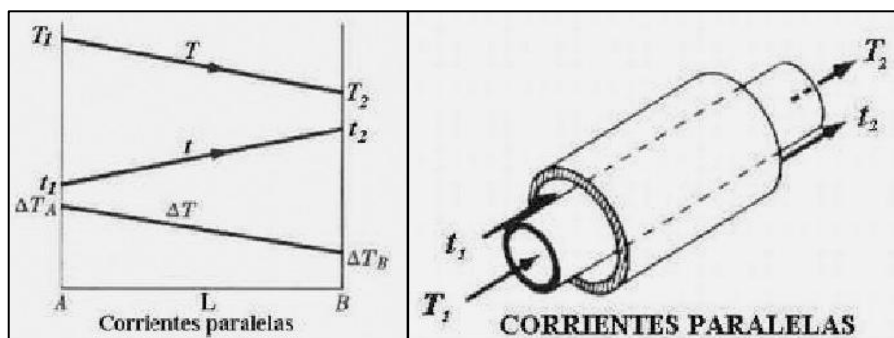


Fig. 2.9 Caso 1. Progresión de las temperaturas y esquema para disposiciones de flujos a corrientes paralelas

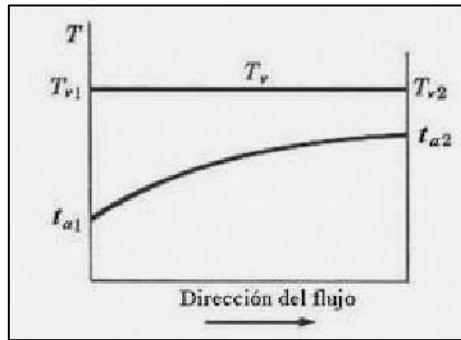


Fig. 2.10 Caso 2. Progresión de las temperaturas

En cualquiera de los dos casos la variación de una o ambas temperaturas acostumbra a no ser lineal como ya observaremos en la sección de resultados del intercambiador de calor.

Enfocando este tipo de dispositivos al mundo de la aeronáutica señalamos que no se suelen utilizar directamente en las corrientes principales del fluido del motor en sí mismo, dado que supondría un aumento de peso para el propio motor considerable. No obstante, podemos ver motores de turbina de gas con intercambiadores para mejorar la eficiencia del ciclo. Éstos, no suelen utilizarse en el ámbito aeronáutico, sino más bien en el ámbito industrial para la generación eléctrica. Además, los intercambiadores trabajan mejor durante largos periodos de funcionamiento y en el caso hipotético del motor de una aeronave, tantos ciclos de arranque y parada acabarían desgastando el intercambiador rápidamente. Esto no quiere decir que las turbinas de gas convencionales no incorporen intercambiadores de calor. Se suelen encontrar en los sistemas de lubricación y combustible. En este caso, los intercambiadores no resultan demasiado pesados al trabajar con corrientes de bajo caudal y fluidos con coeficientes globales de intercambio elevados.

2.3.2. Grado de modelización 0

En este grado de modelización se trata de reproducir un intercambiador de calor ideal con las siguientes hipótesis:

1. El intercambio de calor entre las dos corrientes se produce con una efectividad de intercambio del 100%.
2. Todo el dispositivo trabaja en condiciones adiabáticas con el entorno.
3. Suponemos que los dos fluidos que intercambian calor son exactamente iguales y comparten el mismo calor específico.
4. El flujo másico que entra por el conducto de lado frío es igual al flujo másico que entra por el conducto del lado caliente.

Además, consideramos la hipótesis 3 de la sección 2.1.2. Las ecuaciones utilizadas en este grado de modelización son:

$$(T_{1,in} - T_{1,out}) = (T_{2,out} - T_{2,in}) \quad (2.27)$$

$$\text{Caso cp:} \quad T_{1,out} = T_{2,out} \quad (2.28)$$

$$\text{Caso cc:} \quad T_{1,out} = T_{2,in} \quad T_{2,out} = T_{1,in} \quad (2.29)$$

Donde $T_{1,in}$ es la temperatura de entrada del lado caliente del intercambiador, $T_{1,out}$ la temperatura de salida del lado caliente, $T_{2,out}$ la temperatura de salida del lado frío y $T_{2,in}$ la temperatura de entrada del lado frío.

Debemos señalar, que la hipótesis de la efectividad de transferencia de calor ideal varía en función de las disposiciones de los flujos que tengamos. Para un intercambiador a contracorriente, alcanzaremos la efectividad máxima siguiendo la ecuación 2.29, mientras que para corrientes paralelas, la efectividad máxima se alcanzara siguiendo la ecuación 2.28.

2.3.3. Grado de modelización 1

En este grado de modelización, como en los demás grados de nivel 1 de este trabajo, se han introducido una serie de constantes para simular las pérdidas, en este caso, del intercambiador de calor.

Para la modelización en este grado nos basaremos en las siguientes hipótesis:

1. La efectividad de la transferencia de calor es regulada a través de una constante.
2. La pérdida de presión por cada conducto es estimada a través de valores constantes típicos en un intercambiador de calor.
3. El flujo másico que entra por el conducto de lado frío no tiene por qué ser igual al flujo másico que entra por el conducto del lado caliente.

Además de estas hipótesis adoptamos las realizadas en el punto 3 y 4 del apartado 2.1.2 sobre los valores de los calores específicos y las realizadas en el puntos 2 y 3 del apartado 2.3.2.

Para la modelización de las anteriores hipótesis se han utilizado las siguientes ecuaciones:

$$\dot{m}_1 cp_1 (T_{1,in} - T_{1,out}) = \dot{m}_2 cp_2 (T_{2,out} - T_{2,in}) \quad (2.30)$$

$$\eta_e = \left(1 - \left(\frac{T_{1,out} - T_{2,in}}{T_{1,in} - T_{2,in}} \right) \right) \quad (2.31)$$

$$cp_1 = cp_2 \quad (2.32)$$

$$\eta_{d,h} = \frac{P_{1,in} - P_{1,out}}{P_{1,in}} \quad (2.33)$$

$$\eta_{d,c} = \frac{P_{2,in} - P_{2,out}}{P_{2,in}} \quad (2.34)$$

Donde \dot{m}_1 es el flujo másico del lado caliente del intercambiador, \dot{m}_2 el flujo másico del lado frío del intercambiador, η_e la efectividad del intercambiador, $\eta_{d,h}$ la pérdida de presión del lado caliente, $\eta_{d,c}$ la pérdida de presión del lado frío, cp_1 el calor específico del lado caliente del intercambiador y cp_2 el calor específico del lado frío del intercambiador.

La constante que se encarga de regular la efectividad del intercambiador tendrá diferentes valores para disposiciones a contracorriente o a corrientes paralelas. Para contracorriente, obtendremos una efectividad máxima cuando la resolución de la ecuación 2.31 sea igual a la unidad, mientras que para corrientes paralelas obtendremos una efectividad máxima para un valor de 0.5.

2.3.4. Grado de modelización 2

En este grado de modelización se ha tratado de simular las pérdidas de presión de cada uno de los conductos a partir de la cantidad de calor que es transferido a estos. Además, se ha introducido una distinción entre intercambiadores de calor a contracorriente y a corrientes paralelas para el cálculo de las áreas de intercambio.

Las hipótesis realizadas en este grado son las siguientes:

1. La pérdida de presión es estimada en función del número de Mach y de la cantidad de calor intercambiado para cada conducto.
2. La constante de efectividad del intercambiador es determinada a partir del área de intercambio del dispositivo.
3. Suponemos la utilización de un intercambiador de haz de tubos y coraza.
4. Consideramos un coeficiente global de intercambio constante para una corriente de vapor y una corriente de gases.
5. Los dos fluidos que intercambian calor pueden tener diferentes calores específicos siempre y cuando sean gases.

Además de estas hipótesis, también debemos considerar las vistas en el punto 3 de la sección 2.1.2, el punto 2 de la sección 2.3.2 y el punto 1 de la sección 2.2.5. Este último está relacionado con el tipo de pérdidas de presión que supondremos en los conductos. Las ecuaciones empleadas para este grado se corresponden con la 2.30 y 2.31 para la estimación de las temperaturas de salida del lado frío y el lado caliente. Además, para la estimación de las pérdidas de presión por adición de calor se utilizan las ecuaciones 2.21, 2.22, 2.23, 2.24, 2.25 y 2.26 para cada uno de los dos conductos.

Para el cálculo del área de intercambio de calor y la efectividad y viceversa se utilizan las ecuaciones:

$$A = \frac{\dot{m}_1 c p_1 (T_{1,in} - T_{1,out})}{U \Delta T} \quad (2.35)$$

$$MLDT_{cp} = \frac{(T_{1,in} - T_{2,in}) - (T_{1,out} - T_{2,out})}{\ln\left(\frac{T_{1,in} - T_{2,in}}{T_{1,out} - T_{2,out}}\right)} \quad (2.36)$$

$$MLDT_{cc} = \frac{(T_{1,in} - T_{2,out}) - (T_{1,out} - T_{2,in})}{\ln\left(\frac{T_{1,in} - T_{2,out}}{T_{1,out} - T_{2,in}}\right)} \quad (2.37)$$

$$\Delta T = MLDT \quad (2.38)$$

Donde A es el área de intercambio, U es el coeficiente global de intercambio, $MLDT_{cp}$ es la temperatura media logarítmica para corrientes paralelas y $MLDT_{cc}$ la temperatura media logarítmica para contracorriente.

En este caso, dado que partimos de la hipótesis de que utilizamos un intercambiador de haz de tubos y coraza empleamos la ecuación 2.38. Además para determinar el valor del coeficiente global de intercambio utilizamos una constante típica de la transferencia de calor entre vapor y gases. Concretamente utilizaremos un valor de $0.1163 \text{ kJ/m}^2 \text{ } ^\circ\text{C s}$. En el anexo A2 podemos encontrar una tabla con los valores típicos de este coeficiente.

2.3.5. Programación del módulo

Para la programación de este módulo se han encontrado una serie de dificultades relacionadas con las estimaciones iniciales. Ya que, si los valores de temperatura de salida distan mucho de los correctos, al encontrarnos en la ecuación 2.36 y 2.37 con un logaritmo, podemos encontrarnos con la imposibilidad de resolver el sistema. Además, al utilizar las ecuaciones de Rayleigh nos hemos encontrado con las mismas dificultades que en el punto 2.2.5 de este trabajo.

CAPÍTULO 3. OTROS COMPONENTES

3.1. Atmosfera estándar

3.1.1. Introducción a atmosfera estándar

Para obtener una buena estimación de las actuaciones del motor a diferentes condiciones de vuelo, resulta fundamental tener una descripción de las características principales de la atmosfera real.

La atmosfera real es un sistema muy complejo, compuesto de un gran número de variables y factores, por lo que, intentar describirla en excesivo detalle resultaría muy complejo y poco práctico. Por esta razón se ha tratado de modelar una atmosfera ideal, que represente los valores promedio de la atmósfera real y no represente un gran esfuerzo calcularla. Esta es conocida como la atmosfera estándar.

La atmósfera estándar internacional (ISA, International Standard Atmosphere) constituye un marco de referencia para el mundo aeronáutico. Gracias a este estándar, podemos proporcionar un referente de valores termodinámicos a cualquier altitud.

Dentro de esta atmósfera podemos diferenciar dos regiones atmosféricas: las zonas isotermas y las de gradiente constante de temperatura. En la figura 3.1 podemos observar una ilustración y una tabla sobre las diferentes medidas de temperatura en función de la altitud.

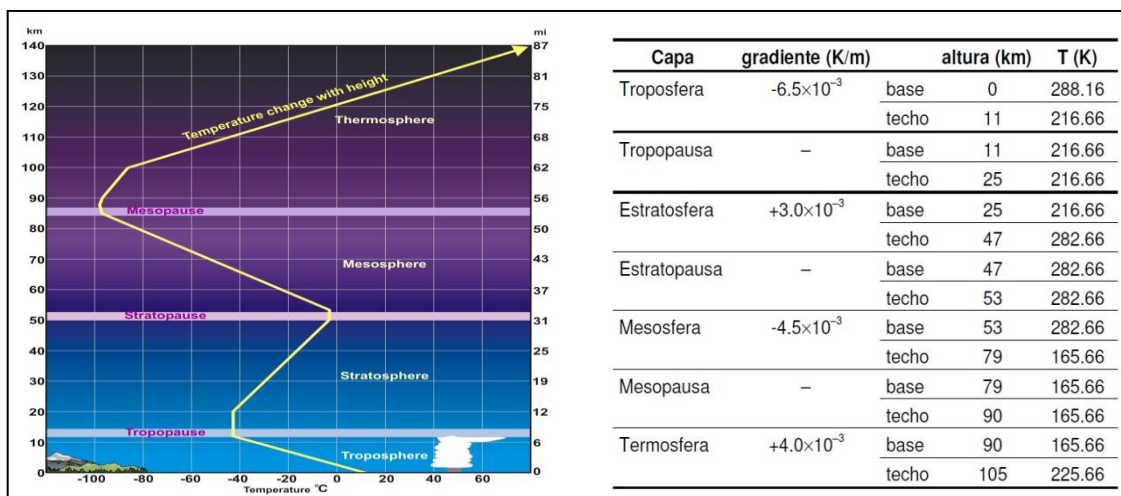


Fig. 3.1 Representación de los valores de temperatura de la atmosfera estándar a diferente altitud

3.1.2. Grado de modelización 0

La idea para este grado de modelización es la de crear un modelo de atmosfera estándar para cualquier altitud hasta 105 Km de altura. Todo ello siguiendo las siguientes hipótesis iniciales:

1. Los gases de la atmosfera están totalmente desprovistos de humedad y polvo
2. La atmosfera gira solidariamente con la superficie terrestre
3. El sistema se encuentra en equilibrio hidrostático
4. La ecuación empleada es referida a la de un gas ideal.

Para la implementación de este grado en concordancia con las hipótesis iniciales hemos utilizado las siguientes ecuaciones:

$$\frac{P}{P_0} = \left(\frac{T}{T_0} \right)^{-\frac{g_0}{R \cdot a_i}} \quad (3.1)$$

$$\frac{P}{P_0} = e^{-\frac{g_0}{R \cdot T}(h-h_0)} \quad (3.2)$$

Donde P representa la presión a una altitud determinada, T la temperatura a esa misma altitud, R la constante de los gases ideales, a_i el gradiente de temperatura, g_0 la aceleración de la gravedad y h la altitud actual a la que queremos realizar el cálculo.

La ecuación 3.1 es utilizada para el cálculo de los valores termodinámicos de la atmosfera en las regiones de gradiente constante de temperatura. Mientras que la ecuación 3.2 es utilizada para el cálculo en las regiones isotermas. En el apartado de programación del módulo 3.1.5 hablaremos más en detalle sobre el procedimiento para realizar el cálculo de estos parámetros en las diferentes regiones de la atmosfera.

Como podemos observar en las ecuaciones 3.1 y 3.2, tenemos una serie de parámetros iniciales (P_0, T_0, h_0). Estos parámetros irán variando en función de la región de la atmosfera de la que se parta. No obstante, para el cálculo de la primera región de gradiente constante se necesitan conocer estos valores a nivel del mar. En este caso tenemos:

$$P_{sl} = 101325 \text{ Pa} \quad (3.3)$$

$$\rho_{sl} = 1.225 \text{ Kg/m}^3 \quad (3.4)$$

$$T_{sl} = 288.16 \text{ K} \quad (3.5)$$

3.1.3. Grado de modelización 1

En este grado de modelización se siguen realizando las mismas hipótesis, se utilizan las mismas ecuaciones y se suponen las mismas condiciones iniciales que en el grado de modelización 0 (apartado 3.1.2 de este trabajo). La única diferencia radica en la inserción de las siguientes ecuaciones para el cálculo de diferentes parámetros:

$$v_s = \sqrt{\gamma * R * T} \quad (3.6)$$

$$M = v/v_s \quad (3.7)$$

$$P_d = \frac{\rho * v^2}{2} \quad (3.8)$$

$$P_t = P_s + P_d \quad (3.9)$$

Donde v_s es la velocidad del sonido local, v la velocidad actual de la aeronave, γ la relación de calores específicos, R la constante de los gases ideales, T la temperatura actual, M el número de Mach, P_d la presión dinámica, P_s la presión estática, P_t la Presión total y ρ la densidad.

En el lugar de la presión estática y los valores de temperatura y densidad, introduciremos los valores calculados a partir de las ecuaciones 3.1 y 3.2. Es decir, se obtendrá un modelo de atmosfera, al que introduciendo valores de altitud y velocidad se podrán obtener los valores correspondientes de temperatura, presión (total y dinámica), densidad y número de Mach.

3.1.4. Grado de modelización 2

La idea para este grado de modelización es la de, siguiendo las ecuaciones y las hipótesis del grado de modelización 1 de la atmosfera estándar (sección 3.1.3), introducir una serie de correcciones para que el cálculo de los parámetros termodinámicos a una determinada altitud de vuelo para que sean lo más fidedignos a la realidad posible.

En concreto hemos introducido correcciones de la aceleración de la gravedad en función de la altitud y la latitud a la que esté volando el motor y correcciones de densidad en función de la humedad relativa del aire. Esta serie de correcciones ha traído consigo la utilización de las siguientes ecuaciones:

$$g^* = G * \frac{M_t}{(R_t + h)^2} \quad (3.10)$$

$$ac = R_t * \cos(\phi) * \omega \quad (3.11)$$

$$g = \sqrt{(g^*)^2 + ac^2 - (2g^*ac \cos(\phi))} \quad (3.12)$$

$$E_s = C_0 * 10^{\frac{C_1 * T}{C_2 + T}} \quad (3.13)$$

$$P_v = RH * E_s \quad (3.14)$$

$$\rho = \left(\frac{P}{R * T} \right) * \left(1 - \frac{0.378 * P_v}{P} \right) \quad (3.15)$$

Donde G representa la constante de gravitación universal, M_t la masa de la tierra, R_t el radio de la tierra, h la altitud actual, T la temperatura actual, ϕ la latitud a la que se encuentra nuestra referencia, ω la velocidad angular de la tierra, C_0, C_1, C_2 son constantes, E_s la saturación de presión de vapor de agua, P_v la presión de vapor de agua, RH la humedad relativa y ρ la densidad.

Las ecuaciones 3.10, 3.11 y 3.12 son las que se utilizan para la corrección de la aceleración de la gravedad en función de la latitud y la altitud a la que se encuentra nuestro motor. Mientras que las ecuaciones 3.13, 3.14 y 3.15 se corresponden con el algoritmo del cálculo de la corrección de la densidad por la humedad relativa. En este se han utilizado una serie de constantes C_0, C_1, C_2 que se corresponden con los valores 6.1078, 7.5 y 237.3 respectivamente. Este algoritmo se basa en la aproximación de la utilización de la ecuación 3.14 en la que la presión de vapor de agua es directamente proporcional al producto de la humedad relativa y la presión de saturación de vapor. No obstante, sería más preciso calcular esta presión de vapor a través del punto de rocío, pero resulta menos intuitivo que la humedad relativa. [9,11]

3.1.5. Programación del módulo

Este módulo, a diferencia del tipo de programación de los que componen los módulos de transferencia de calor, está diseñado para que solo funcione en un sentido, es decir, solo será capaz de resolver una serie de incógnitas específicas para unos parámetros de entrada determinados. Aun así, la programación de este módulo ha resultado ser una tarea engorrosa, ya que, la atmosfera estándar, hasta los 105 Km de altura, está compuesta por cuatro regiones a gradiente constante y tres por regiones isotermas. Debido a esto, resulta necesario elaborar una serie de casos para cada región en la que se quiera calcular los valores termodinámicos. Además, para cada ecuación se requieren los valores P_0, T_0 o h_0 y esto hace que sea imprescindible realizar el cálculo a nivel del mar hasta la región en la que estemos en cada caso. El resultado final trae consigo una repetición de parte del código para las regiones más altas de la atmosfera, aunque de sencilla implementación y simple funcionamiento.

3.2. Propiedades del fluido

3.2.1. Acerca de las propiedades del fluido

El objetivo de este módulo es el de considerar los diferentes calores específicos para los diferentes cambios de temperatura que sufre el fluido a lo largo de todo su recorrido por el motor.

3.2.2. Grado de modelización 0

En este grado se ha tratado de simplificar al máximo la asignación de valores del calor específico a presión constante (cp) y la relación de calores específicos (γ). Por este motivo se ha escogido una constante representativa para un fluido considerado como aire para todo el recorrido en el motor y que posee una constante $R_{air} = 0.287 \text{ kJ/kgK}$. Por lo tanto, el calor específico a presión constante tendrá un valor de 1.005 kJ/kgK y la relación de calores específicos tendrá un valor de 1.40.

3.2.3. Grado de modelización 1

En este grado de modelización se han tenido en cuenta 2 tipos de fluidos representativos a lo largo de su recorrido por el motor, antes y después del proceso de combustión. Antes de que el fluido sufra el proceso de combustión, hemos escogido los valores representativos un calor específico (cp) de 1.005 kJ/kgK y una relación de calor específicos (γ) de 1.40, igual en la modelización 0, sección 3.2.2. Después de que el fluido sufra el proceso de combustión hemos escogido unos valores de $cp = 1.148 \text{ kJ/KgK}$ y una $\gamma = 1.333$. Es valores han sido extraídos de la referencia [2].

3.2.4. Grado de modelización 2

En este grado de modelización se han tratado encontrar unos valores determinados de cp y γ en función de la temperatura y de la relación de combustible/aire de la cámara de combustión. En el apartado de programación de este módulo se comentarán cual ha sido el proceso de aproximación de estas curvas.

3.2.5. Programación del módulo

La mecánica de funcionamiento de este módulo es muy similar a la de la atmosfera estándar. A través del paso de una serie de parámetros iniciales, en este caso, de temperatura y la relación de combustible/aire, obtenemos una serie de valores de c_p y γ . La única dificultad que ha planteado este módulo, a la hora de implementarlo en código de programación, es la de conseguir modelar una superficie para que, a cualquier valor de temperatura y relación de combustible/aire (f), podamos obtener los valores correctos de c_p y γ . Para ello, se ha implementado las siguientes ecuaciones por partes.

Para el valor de c_p :

$$c_p = 1.973 * 10^{-4}T + 0.9448 + \begin{cases} f * 0.7714 & \text{para } 300 < T \leq 1000 \\ f * 3.7081 & \text{para } 1000 < T \leq 1800 \end{cases} \quad (3.16)$$

Y para los valores de γ :

$$\gamma = -f * 0.7407 + \begin{cases} -8.2571 * 10^{-5}T + 1.4222 & \text{para } 200 < T \leq 1000 \\ -6.2125 * 10^{-5}T + 1.4012 & \text{para } 1000 \leq T < 1800 \end{cases} \quad (3.17)$$

En la figura 3.2 podemos observar una representación de las curvas que relacionan los calores específicos y la relación de calores específicos con la temperatura y la relación de combustible/aire y la aproximación realizada por estas ecuaciones por partes.

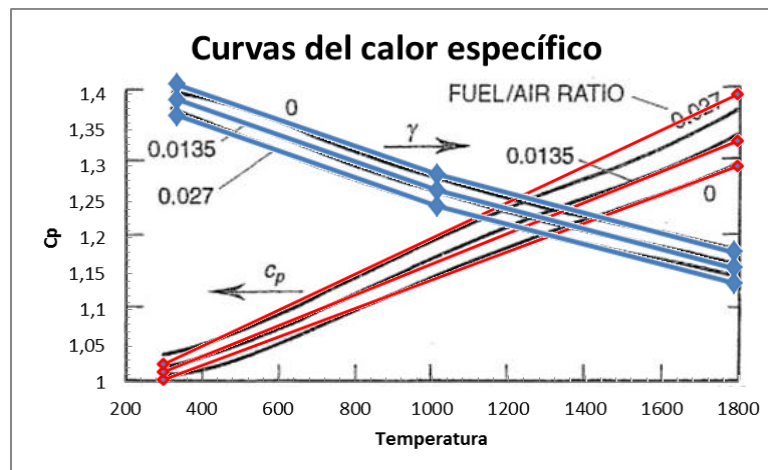


Fig. 3.2 Modelización de las curvas de calor específico. Las líneas rojas se refieren a la aproximación realizada a través de la ecuación por partes 3.16. Las líneas azules se refieren a la aproximación realizada a través de la ecuación por partes 3.17. Las líneas negras se refieren a la curva real

3.3. Parámetros del motor

3.3.1. Introducción a parámetros del motor

El objetivo de este módulo consiste en calcular una serie de parámetros de rendimiento y características generales del motor una vez ensamblado.

3.3.2. Grado de modelización 0

Para este único grado de modelización, simplemente, se han introducido una serie de ecuaciones que dan como resultado las características fundamentales del motor. Estas ecuaciones están compuestas por:

$$F = \dot{m}(v_{out} - v_{in}) + (A(P_{out} - P_{in})) \quad (3.18)$$

$$\dot{W}_p = F * v_a \quad (3.19)$$

$$\dot{Q}_{in} = (f * q_{fuel}) * \dot{m} \quad (3.20)$$

$$\eta_p = \frac{\dot{W}_p}{\dot{Q}_{in}} = \frac{2}{1 + (v_{out}/v_{in})} \quad (3.21)$$

$$\eta_e = \frac{\dot{m}(v_{out}^2 - v_{in}^2)/2}{\dot{Q}_{in}} \quad (3.22)$$

$$\eta_o = \eta_e * \eta_p \quad (3.23)$$

$$FS = \frac{F}{\dot{m}} \quad (3.24)$$

$$TSFC = \frac{f}{FS} * 10^6 [g/KN * s] \quad (3.25)$$

$$PSFC = \frac{f}{W_{shaft}} [Kg/Kw * s] \quad (3.26)$$

Donde F es el empuje que desarrolla el motor, \dot{m} el flujo másico de entrada, v_{out} la velocidad de los gases de salida, v_{in} la velocidad de los gases de entrada, A el área transversal del motor, P_{out} la presión desarrollada de salida, P_{in} la presión atmosférica, \dot{W}_p la potencia de propulsión, v_a la velocidad de nuestro motor, \dot{Q}_{in} la calor transferida al fluido, f la relación combustible/aire, q_{fuel} el poder calorífico del combustible, η_p la eficiencia de propulsión, η_e la eficiencia de la conversión de energía, η_o la eficiencia global, FS es el empuje específico, W_{shaft} es la potencia que requiere la hélice en caso de tenerla, $TSFC$ el consumo específico de combustible por unidad de empuje y $PSFC$ el consumo específico de combustible por unidad de potencia.

CAPÍTULO 4. MÉTODO DE RESOLUCIÓN NEWTON-RAPHSON

4.1. Métodos de resolución de ecuaciones no lineales

4.1.1. Base teórica

En esta sección se comentarán los diferentes métodos para la resolución de las ecuaciones no lineales que podrían ser utilizados en nuestro sistema motor, o incluso dentro de cada uno de los módulos de transferencia de calor incluidos en este trabajo. Una vez hayamos comentado estos métodos se definirá cual es el escogido para nuestro programa y sus ventajas principales respecto al resto.

Cualquier ecuación se puede expresar de forma que $f(x) = 0$. Estas son precisamente la idea para esta clase de métodos, tratar de encontrar los ceros. En este caso las veremos para funciones reales de unas variables reales. En algunos casos, podemos encontrar la solución de forma analítica rápidamente, pero para ecuaciones más complicadas es necesaria la utilización de métodos numéricos asistidos por computación.

Pese a que una solución numérica no deja de ser una solución aproximada, podemos escoger el grado de precisión y adecuarlo a nuestras necesidades. Además, este tipo de soluciones suelen ser más rápidas de encontrar y suelen ser bastante acertadas para un grado de precisión adecuado. Por este motivo, y por el hecho de que, en nuestro sistema motor, o simplemente dentro de nuestros módulos, encontramos sistemas enteros de ecuaciones no lineales, se hace imprescindible la utilización de estos métodos para encontrar las soluciones del sistema. En las siguientes secciones se describirán los principales métodos de resolución numérica.

4.1.2. Principales métodos de resolución

En la actualidad podemos encontrar varios métodos de resolución de ecuaciones no lineales adaptables a la programación. A continuación los nombraremos e indicaremos cual es el que más se ajusta a nuestras necesidades y porque. Entre ellos encontramos:

- El método de la bisección.
- El método de *Regula falsi*.
- El método de la iteración de punto fijo.
- El método de la secante.
- El método de Newton.
- El método de Muller.

Los métodos de la bisección y *Regula falsi* son métodos de dominio cerrado bastante robustos, pero tienen el inconveniente de converger lentamente. Por este motivo se descartan rápidamente, ya que necesitamos un método con un ratio de convergencia elevado.

Los métodos restantes son de dominio abierto. En el caso del método de la iteración de punto fijo posee el inconveniente de converger de forma lineal o incluso no encontrar la solución para $|g'(\alpha)| > 1.0$, por lo que no es recomendable su utilización.

Por último tenemos los métodos de la secante, de Newton y de Muller. La principal ventaja del método de la secante y de Muller es que no necesitan evaluar el cálculo de la derivada, a diferencia del método de Newton. No obstante, el ratio de convergencia del método de Newton es cuadrático, y en los anteriores métodos el ratio de convergencia para el método de la secante y de Muller es de 1.62 y 1.84 respectivamente. Además, el problema del cálculo de la derivada para el método de Newton lo podemos solventar a través de una aproximación como ya veremos en las siguientes secciones. Por estos motivos, hemos optado por implementar el método de Newton-Raphson. Para obtener más información acerca de estos métodos de resolución y sus características, mirar la referencia [3].

4.1.3. Método de Newton-Raphson

Este método soluciona la ecuación $f(\bar{x})$, para una incognita, considerando las tangentes a la curva $f(x)$. Por este motivo, le es necesario conocer la derivada de la función. El proceso de resolución empieza a partir de una estimación inicial de la solución x_1 , para la cual determinamos $f(x_1)$ y encontramos la tangente. Así encontramos la nueva estimación x_2 y mediante iteraciones conseguimos llegar a la solución. Este proceso se puede observar en la figura 4.1.

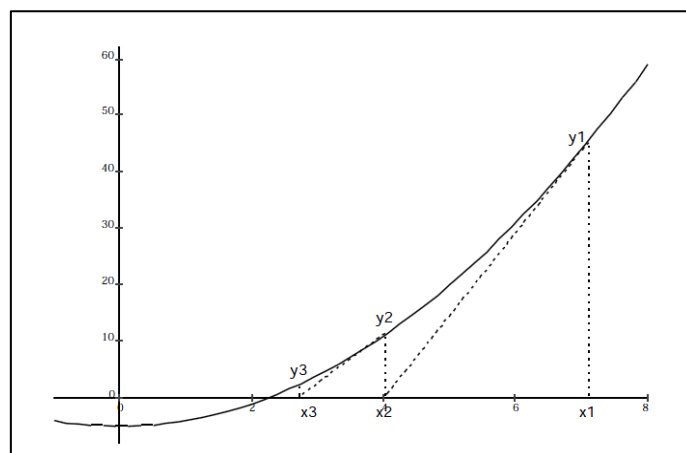


Fig. 4.1 Método de Newton-Raphson

A partir de la gráfica podemos deducir que la expresión analítica de este método se corresponde con:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad (4.1)$$

El método de Newton-Raphson no siempre converge, pero cuando lo hace, la convergencia es más rápida que en otros métodos, ya que la hace de forma cuadrática.

4.2. Programación del método de Newton-Raphson

La idea para este trabajo es la de conseguir simular una serie de módulos que representen las funcionalidades de dispositivos de intercambio de calor. Para llevar a cabo esta tarea, no bastaba con implementar un método de Newton para una variable concreta. El auténtico reto consistía en hacer un diseño en programación capaz de encontrar la solución de cualquier incógnita que se asignara, dando valores a los demás parámetros.

En las siguientes secciones de este capítulo, se comentarán cuáles son los procesos que se emplean para la implementación de este método, y cuales han sido las estrategias que se han seguido para resolver los problemas que han ido surgiendo durante su creación.

4.2.1. Programación del método para una sola variable

La tarea de programar un método de Newton para una sola variable se puede realizar simplemente asignando un caso para cada variable de la ecuación. Como ya hemos visto en la sección 4.1.3, y siguiendo la ecuación 4.1, necesitamos evaluar las funciones $f(x)$ y estimar su derivada para llegar iterativamente a la solución $f(x) = 0$.

Una de las posibles soluciones a la hora de programarlo, era, como ya hemos comentado, asignar un caso para cada variable de la ecuación, calculando por separado la derivada de la función respecto a la incógnita en la que nos encontrásemos. Esta solución resultaba rápida y efectiva únicamente en el caso de trabajar con una sola ecuación. A la hora de trabajar con dos o más, la carga de trabajo programando un caso para cada incógnita de cada ecuación resultaría poco práctica y muy laboriosa. Por este motivo, tal y como se aconseja en la referencia [3] para el cálculo de la derivada, en el método de Newton, esta se aproxima mediante diferencias finitas del siguiente modo:

$$f'(x) = \frac{f(x+\varepsilon) - f(x)}{\varepsilon} \quad (4.3)$$

Dónde ε se toma pequeño en relación al valor de x . En el caso de que ε sea demasiado pequeño, aparecerán errores de redondeo, y en el caso de que sea demasiado grande, el ratio de convergencia disminuirá. Por esta razón deberemos escoger con cautela nuestro valor de ε a la hora de resolver nuestra ecuación o sistema. A este procedimiento se le conoce como "*Approximate Newton method*". Una vez tenemos determinada la derivada ya somos capaces de diseñar un programa que pueda encontrar la solución para cada elemento de la ecuación.

La distribución del programa para realizar el método puede dividirse en 5 puntos:

1. En primer lugar, inicializamos todos los parámetros, asignando el valor de -1 para que el programa pueda reconocer que es una incógnita. Además, introducimos una estimación inicial de nuestra incógnita y asignamos un valor al diferencial de x , que más adelante utilizaremos para el cálculo de la aproximación de la derivada.
2. En segundo lugar, realizamos la derivada de la función a partir de la aproximación expresada en la ecuación 4.3. Para ello utilizaremos los parámetros iniciales, la incógnita con el valor de la estimación inicial y el diferencial de x .
3. Evaluamos la función $f(x)$ para los parámetros iniciales y la incógnita con el valor de la estimación inicial.
4. Aplicamos la ecuación del método de Newton-Raphson (4.2) y extraemos el nuevo valor de x .
5. A partir del nuevo valor de x , mediante una cláusula "while" repetimos los pasos 3 y 4 hasta que el valor de $f(x)$ sea menor al error que creamos oportuno.

Para la implementación de este método resulta fundamental que las llamadas de las funciones de los módulos estén diseñadas de tal forma que devuelvan la evaluación de $f(x)$ para cualquier x que le pasemos.

Ahora, la única restricción es precisamente el hecho de que nuestro motor estará compuesto por un sistema completo de ecuaciones y por lo tanto, deberemos trabajar con matrices. En la siguiente sección se comenta todo este proceso y cuáles son las dificultades añadidas.

4.2.2. Programación del método para un sistema multivariable

La programación del método de Newton para un sistema de ecuaciones sigue el mismo procedimiento que en el caso de una sola variable, sólo que en vez de trabajar con una única función, trabajaremos con una matriz de funciones.

Durante el desarrollo del programa para este caso, se han encontrado dos problemas fundamentales: el cálculo de la matriz de derivadas parciales y la resolución de la ecuación de Newton 4.1 para matrices.

En primer lugar, al trabajar con funciones vectoriales de múltiples variables, no tendremos una sola derivada, sino que tendremos una matriz de derivadas parciales. Esta es conocida como el Jacobiano. Para su cálculo utilizamos la expresión 4.4.

$$J_F(x_1, \dots, x_n) = \begin{bmatrix} \frac{\delta f_1}{\delta x_1} & \dots & \frac{\delta f_1}{\delta x_n} \\ \vdots & \ddots & \vdots \\ \frac{\delta f_m}{\delta x_1} & \dots & \frac{\delta f_m}{\delta x_n} \end{bmatrix} \quad (4.4)$$

El problema del cálculo del Jacobiano radica en como poder realizarlo de manera que funcione tanto para sistemas de 2 ecuaciones como para sistemas de 20. En el primer caso podríamos llamar a la aproximación de la derivada cuatro veces, y crear la matriz del Jacobiano de 2x2. Pero en el segundo caso sería muy poco práctico tener que introducir dentro del código 400 llamadas a la aproximación.

Para resolver este problema, en primer lugar se han recogido todos los parámetros de las ecuaciones y se han agrupado en una matriz donde para cada fila tendremos una ecuación distinta y para cada columna tendremos la posición de la variable en el orden de llamada de la función.

A continuación se ha utilizado una cláusula del tipo “while” para recorrer sus n columnas. Dentro de esta condición, introduciremos las llamadas a las x funciones del sistema. La construcción de este sistema es mucho más compleja de lo que parece a simple vista, ya que, en el caso de que existan parámetros que se repitan en varias ecuaciones del sistema, este factor se deberá tener en cuenta a la hora de realizar la matriz de derivadas parciales. Por este motivo, antes de crearla, se ha diseñado un filtro que se encarga de extraer y ordenar las incógnitas y sus conexiones agrupándolas en una matriz tridimensional. En esta, las variables agrupadas en “z” se emplearán para guardar el número de incógnitas verdaderas (sin repetición por conexiones). En el caso en que nuestro sistema esté compuesto por doce ecuaciones, en esta variable se deberán almacenar doce incógnitas. En las posiciones “y” y “x” se almacenan la posición del número de la ecuación en la que se encuentran y la posición dentro de nuestra ecuación agrupada por el orden de llamada de la función respectivamente.

Una vez hemos resuelto el Jacobiano utilizando la ecuación de la aproximación de la derivada (4.3) sólo nos queda encontrar una forma para resolver el sistema de matrices que requiere el cálculo de la ecuación de Newton (4.1).

Este problema resulta prácticamente imposible si se trata de resolver mediante las funciones de las librerías básicas que vienen con C++. Nosotros tendremos

una ecuación del tipo $Ax = B$ en el que B representa la matriz de llamadas de la función para $f(x_1, \dots, x_n)$ y A la matriz del jacobiano. Si intentáramos implementarlo con las librerías básicas, nos encontraríamos con una expresión del tipo $x = A^{-1} * B$. Llegados a este punto, nos veríamos obligados a calcular la inversa de la matriz del jacobiano lo cual resultaría una tarea muy laboriosa y nada práctica. Por ello se han hecho uso de las librerías utilizadas en fortran "LAPACK" adaptadas a C++, "CLAPACK", para la resolución del sistema $Ax = B$ mediante la función "dgesv_". Una vez conseguido instalar correctamente las librerías para el compilador de Visual studio 2010, somos capaces de resolver este sistema y con ello el método de Newton-Raphson para un sistema multivariable.

En general, el proceso a seguir para la resolución del método de Newton multivariable es:

1. Inicializamos todos los parámetros, asignando el valor de -1 para que el programa pueda reconocer que es una incógnita. Además, introducimos una estimación inicial para nuestras incógnitas y asignamos un valor al diferencial de x.
2. Indicamos las conexiones que existirán entre las diferentes variables del sistema, para que el programa pueda agrupar las incógnitas verdaderas posteriormente.
3. Realizamos el filtrado y la agrupación de incógnitas verdaderas.
4. Calculamos el Jacobiano.
5. Evaluamos la función $f(x_1, \dots, x_n)$ para los parámetros iniciales y las incógnitas con los valores de la estimación inicial.
6. Resolvemos el sistema $Ax = B$ mediante la función de la librería CLAPACK "dgesv_".
7. Una vez encontrada la matriz x producto del sistema, la introducimos en la ecuación de Newton 4.1
8. Repetimos de forma continuada los pasos 5, 6, 7 hasta que las soluciones del sistema hayan convergido.

En este caso, observamos que sólo realizamos el cálculo del Jacobiano una sola vez. Esto puede traer consigo una serie de problemas, ya que, si la estimación inicial de los parámetros no es muy acertada, podríamos perder nuestra convergencia cuadrática. Por este motivo es conveniente realizar el cálculo del Jacobiano a cada iteración y con ello poder conservar nuestro ratio de convergencia. A este procedimiento se le conoce como "*lagged Newton's method*".

Podemos encontrar en el anexo A5 el código de programación del método de Newton comentado.

CAPÍTULO 5. RESULTADOS DE LA IMPLEMENTACIÓN DE LOS MÓDULOS

5.1. Ratio de convergencia del método de Newton

En esta sección observaremos los diferentes ratios de convergencia que podemos obtener en la utilización de un procedimiento del tipo “*Aproximate Newton method*” y el procedimiento “*Lagged Newton’s method*” mencionados en los apartados 4.2.1 y 4.2.2. Todo ello, en función del error introducido en las estimaciones iniciales para el módulo del sistema de postcombustión.

5.1.1 Resultados del ratio de convergencia para el sistema de postcombustión.

En este caso, para comprobar cuál es el ratio de convergencia, y en general, el método que mejor se comporta, hemos utilizado el módulo del sistema de postcombustión de grado 2. Este, está compuesto por una serie de 14 funciones y está diseñado para que los resultados tengan un error no mayor del 0.0001%. Podemos observar los resultados en las figuras 5.1 y 5.2

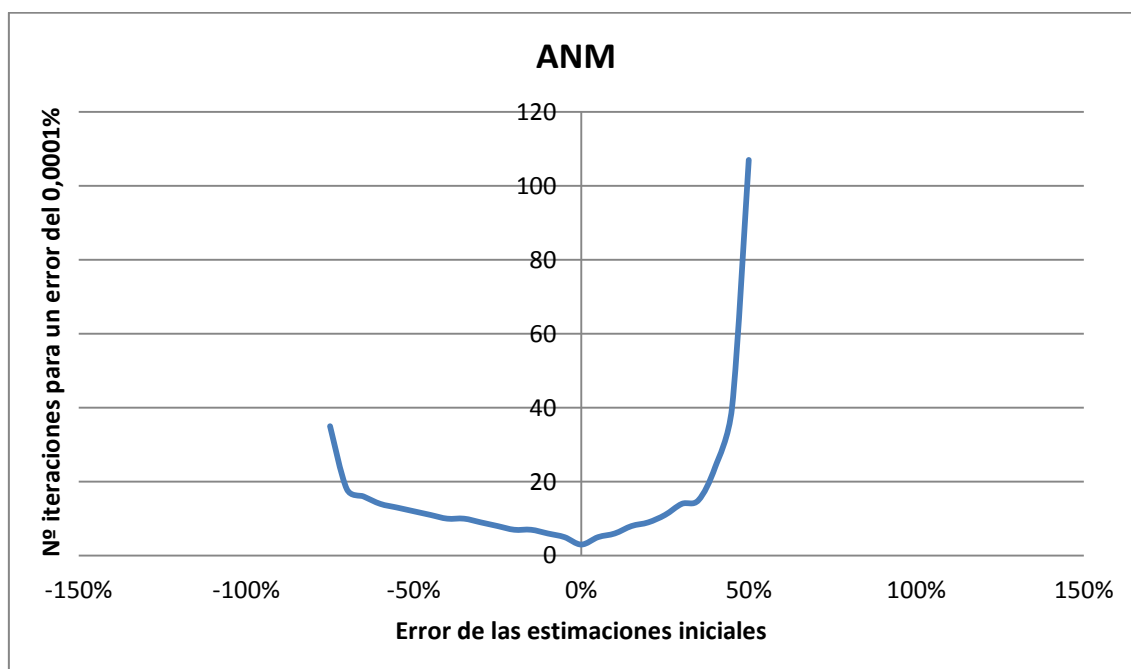


Fig. 5.1 Número de iteraciones en función del error de las estimaciones iniciales para el procedimiento del tipo “*Aproximate Newton method*” (ANM)

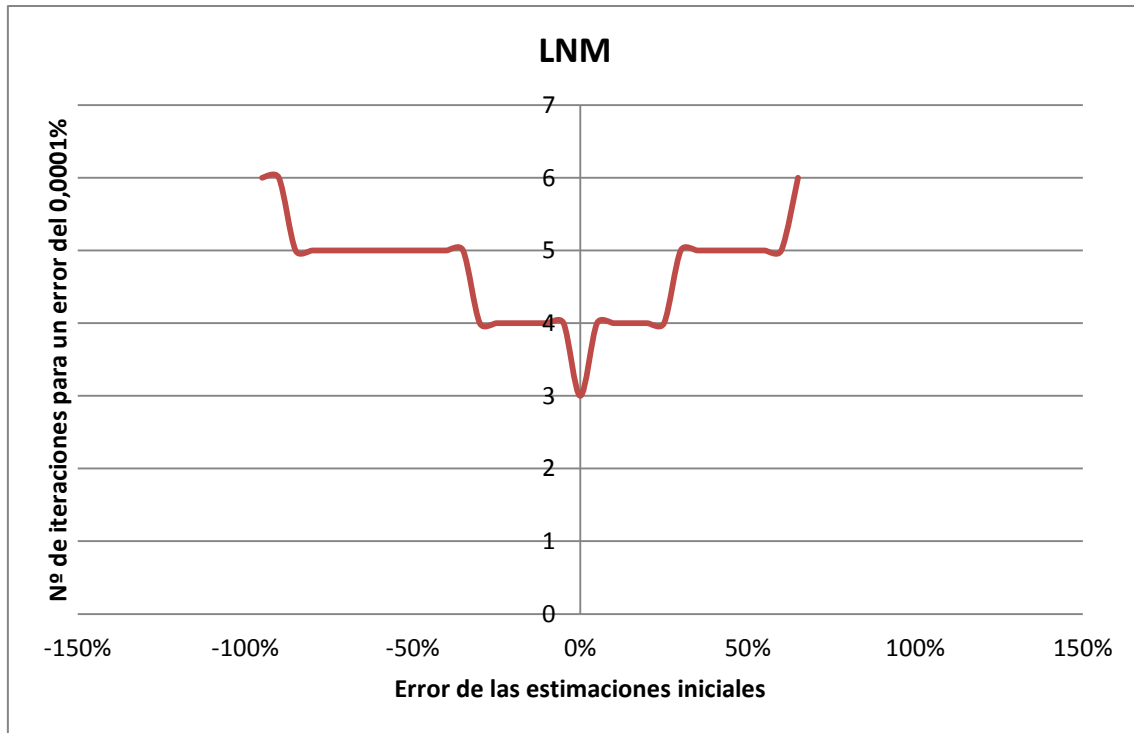


Fig. 5.2 Número de iteraciones en función del error de las estimaciones iniciales para el procedimiento del tipo “*Lagged Newton’s method*” (LNM)

La peculiaridad de este sistema, y factor decisivo a la hora de resolverlo, radica en las ecuaciones 2.24 y 2.25, las cuales al tener una incógnita elevada a una potencia 4, si la estimación inicial se desvía lo suficiente del valor correcto, todo el sistema convergirá en una solución errónea.

Este es el caso para el procedimiento LNM al llegar a unos errores del -100% y +70%. Mientras que en el procedimiento ANM, al llegar a unos errores del 50% y -75%, la solución nunca convergirá. Observamos pues, la criticidad en la elección de las estimaciones iniciales en este último procedimiento para el cálculo del jacobiano.

Podemos concluir, que el procedimiento que más se ajusta a nuestras necesidades y que por tanto, es el que se implementará para el test de los módulos y de los motores ensamblados es el “*Lagged Newton’s method*”.

5.2. Test de prueba de la cámara de combustión

En este apartado se analizarán los diferentes resultados del módulo de grado máximo de la cámara de combustión y se estimarán los posibles errores derivados de las ecuaciones. Además, se compararán los resultados con los de un programa comercial de simulación de motores de turbina, *GasTurb 11*.

5.2.1. Resultados obtenidos en la cámara de combustión

En la cámara de combustión se han obtenido los siguientes resultados para los valores de emisiones. Para unas condiciones iniciales de flujo másico igual a 10 Kg/s, un volumen de combustión de 0.09264 m³ y una eficiencia de combustión del 95% obtenemos los siguientes resultados:

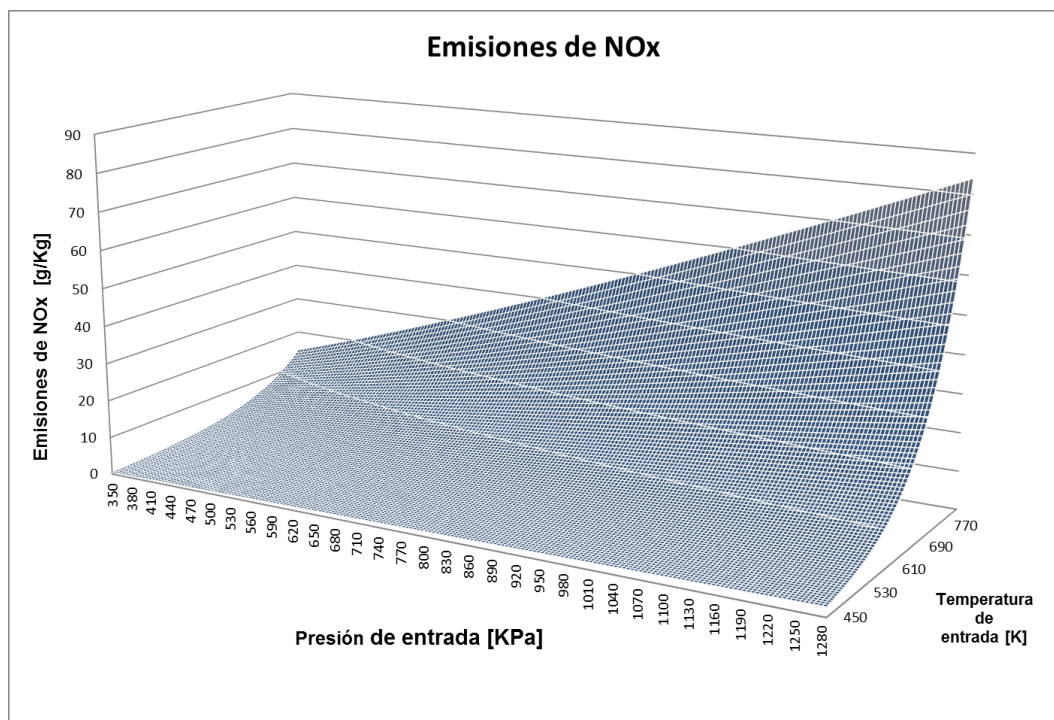


Fig. 5.3 Nivel de emisiones de óxidos de nitrógeno

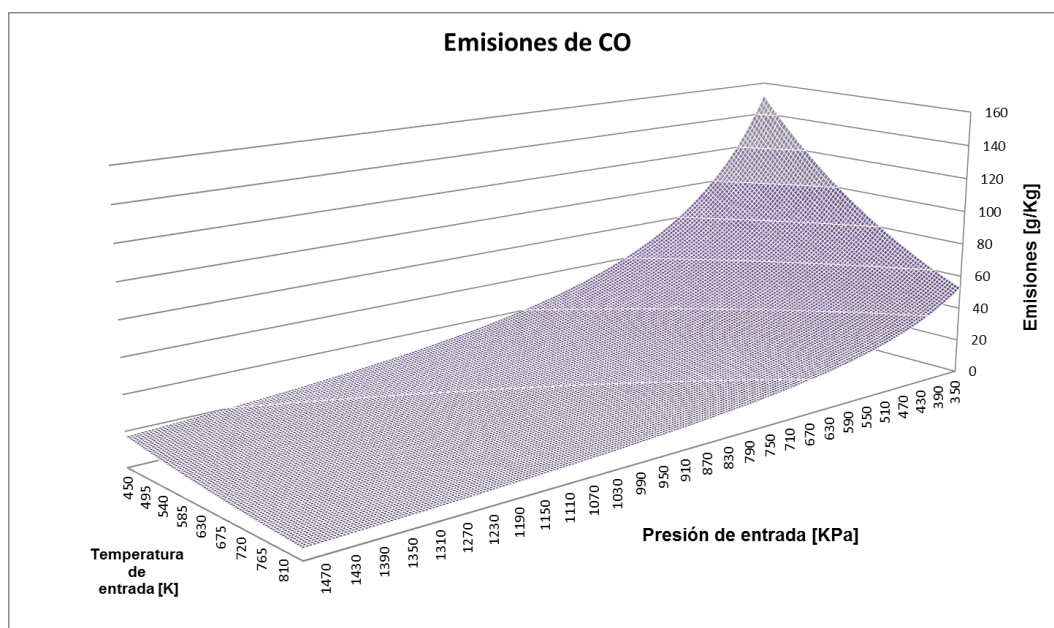


Fig. 5.4 Nivel de emisiones de monóxidos de carbono

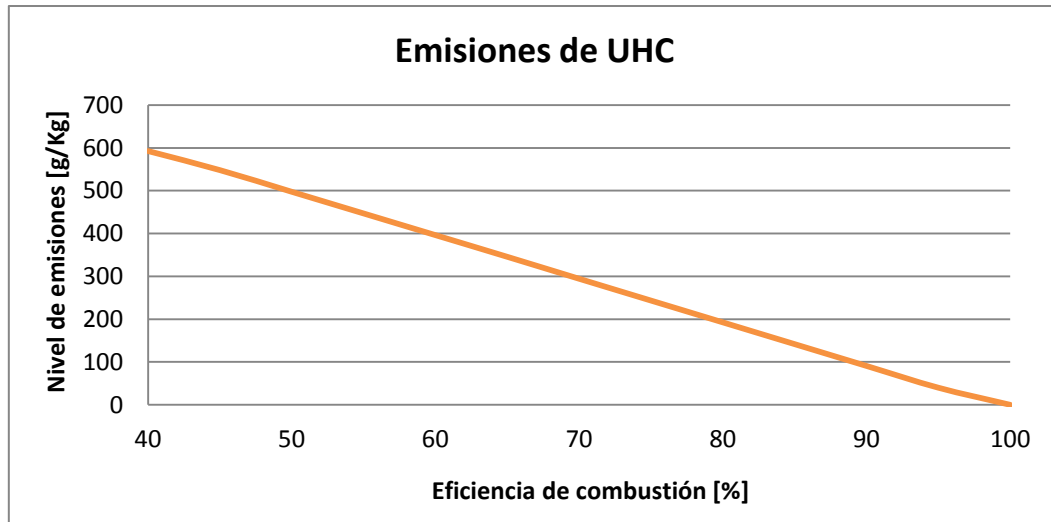


Fig. 5.5 Nivel de emisiones de hidrocarburos no quemados (UHC)

Para poder observar de un modo más significativo en los niveles de emisiones de todos los contaminantes en función de la temperatura en la zona primaria. En este caso tendremos una condiciones iniciales de presión de entrada de 650 kPa, un flujo másico de 20 kg/s, una eficiencia de combustión del 95% y un volumen de combustión de 0.09264 m^3 .

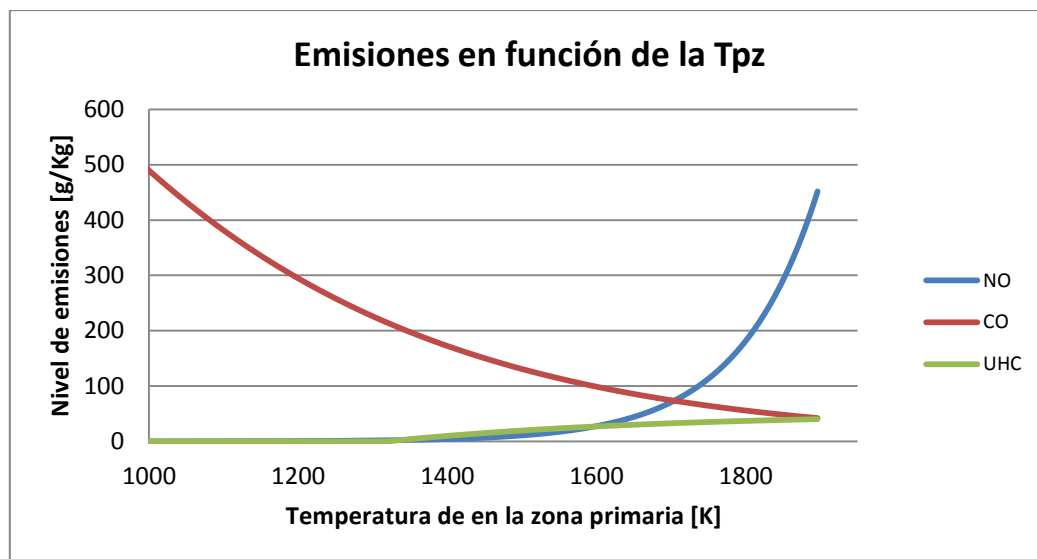


Fig. 5.6 Nivel de emisiones en función de la temperatura en la zona primaria (Tpz)

Podemos concluir en esta sección que el punto de trabajo óptimo de la cámara de combustión para una emisión de contaminantes mínima será para una temperatura en la zona primaria de 1700 K aproximadamente. Este dato se confirma con las gráficas de emisiones adjuntadas en el anexo A3 proporcionadas por la referencia [1].

5.2.2. Análisis de errores y comparación

Uno de los factores más significativos que introducen error en el sistema de la cámara de combustión viene dado por la aproximación realizada para el cálculo del incremento de temperatura comentada en la sección 2.1.6. El error máximo introducido aumenta a medida que nos acercamos a mezclas de aire-keroseno mayores. En este caso el error en el incremento de temperatura máximo ronda los ± 75 K. En la figura 5.7 podemos observar la temperatura de salida de la cámara con esta desviación y la curva correspondiente a la del programa comercial *GasTurb 11*.

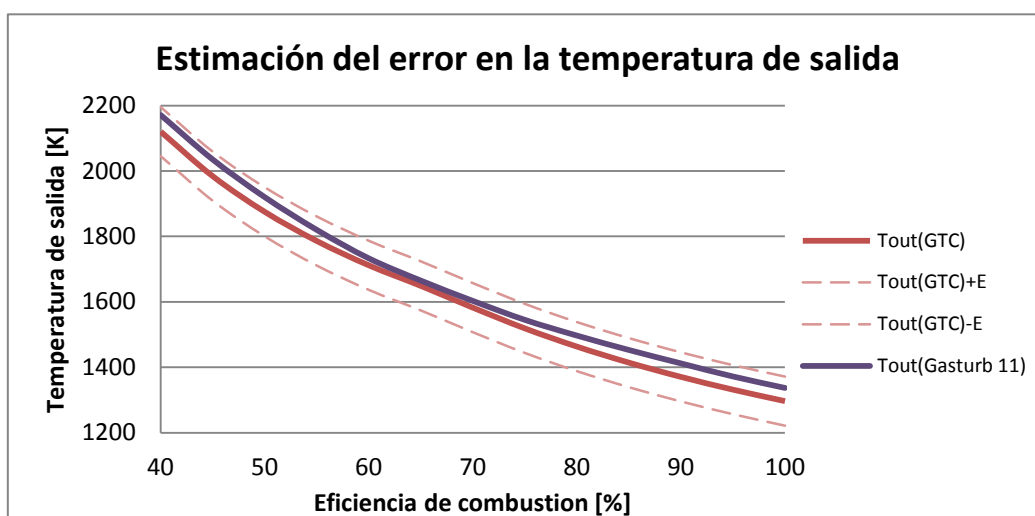


Fig. 5.7 Estimación del error y comparación de la temperatura de salida $T_{out}(GTC)$ (obtenida en nuestro módulo con un valor de error $\pm E$) y la temperatura $T_{out}(GasTurb 11)$ (obtenida por el programa comercial)

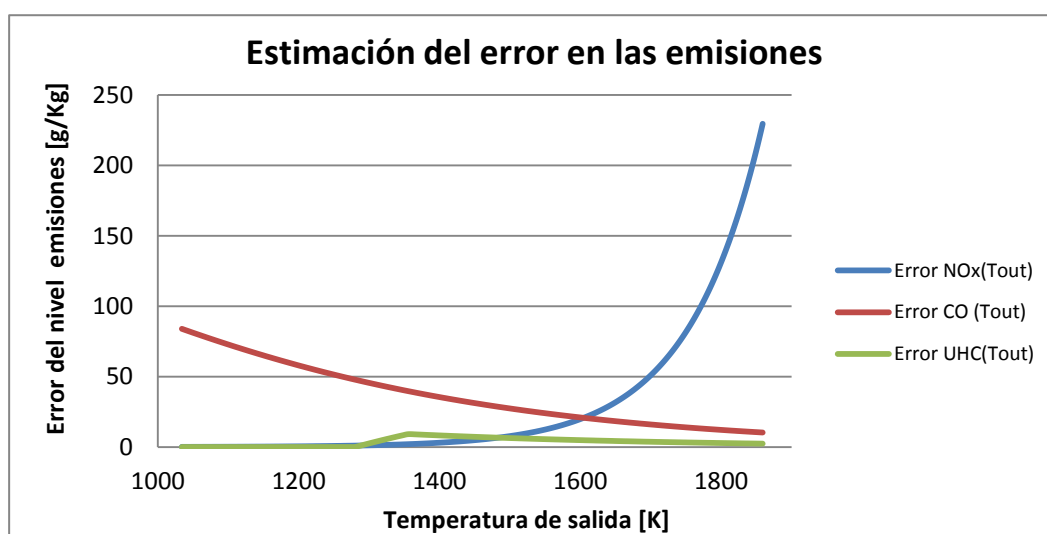


Fig. 5.8 Estimación del error absoluto de emisiones en función de la temperatura de salida

Observamos en la figura 5.7 como las temperaturas de salida de nuestro módulo van bastante parejas a las del programa comercial en comparación con el posible error que sistema podría introducir debido a las aproximaciones de las curvas de la figura 2.3. En la figura 5.8 tenemos una estimación de los errores en las emisiones en función de la temperatura de salida. Podemos observar como para una serie de valores muy reducidos o muy elevados para una cámara de combustión, los posibles errores en las emisiones son considerables, especialmente en los óxidos de nitrógeno para temperaturas elevadas. No obstante, dado que el régimen de funcionamiento para las cámaras de combustión suelen estar entre 1000-1500 K obtendremos un error máximo reducido. Aun así, este no es el único error que se introduce en la estimación de las emisiones. Los parámetros que intervienen en la ecuación 2.15, D_0^2 y λ_{eff} están escogidos a partir de valores representativos para unas condiciones determinadas. Aun así, el error que pueden producir lo consideramos despreciable.

5.3. Test de prueba del sistema de postcombustión

En ese apartado se seguirá la misma metodología a la hora de analizar el componente de la cámara de combustión.

5.3.1. Resultados obtenidos en el sistema de postcombustión

Para el grado de modelización máximo del sistema de postcombustión se introducen una serie de ecuaciones para estimar el cálculo de las pérdidas de presión para un fluido de Rayleigh. En la figura 5.9 podemos encontrar los resultados de esta estimación.

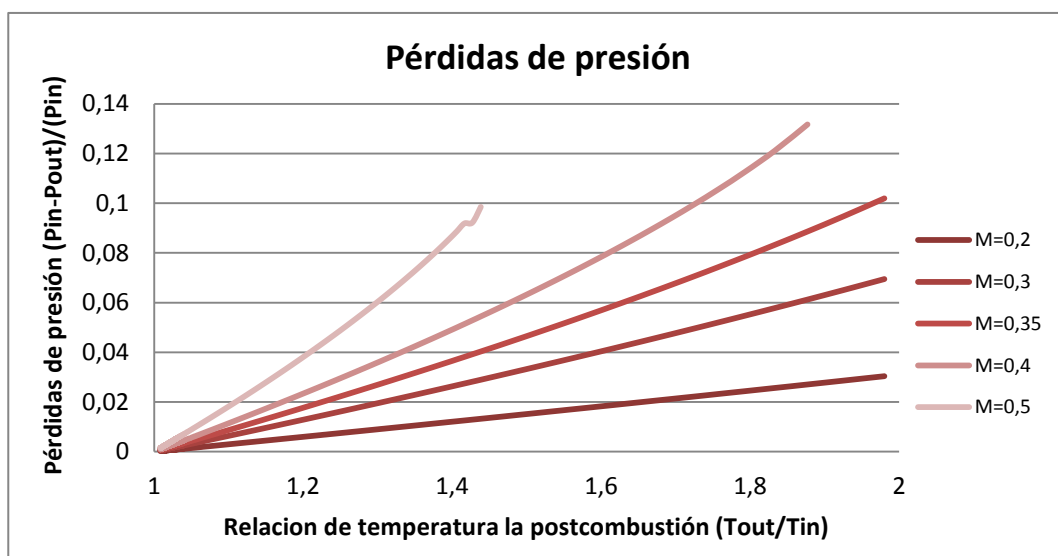


Fig. 5.9 Relación entre las pérdidas de presión experimentales y la relación de temperaturas para diferentes números de Mach de entrada

Podemos concluir con éxito que las dos gráficas (Fig. 2.7 y 5.9) se corresponden y por lo tanto, confirmamos que hemos implementado satisfactoriamente la introducción de una serie de pérdidas de presión en función de la cantidad de calor introducido y el Mach de entrada.

5.3.2. Análisis de resultados y comparación

Como ya se ha comentado en la sección 5.2.2 el factor que más error puede introducir a nuestro sistema es el referente a la aproximación de las curvas de incremento de temperatura para mezclas de keroseno-aire. Para unas condiciones iniciales de presión de entrada igual a 360.27 kPa, una temperatura de entrada de 846.34 K y un flujo másico de 28.8 Kg/s obtenemos:

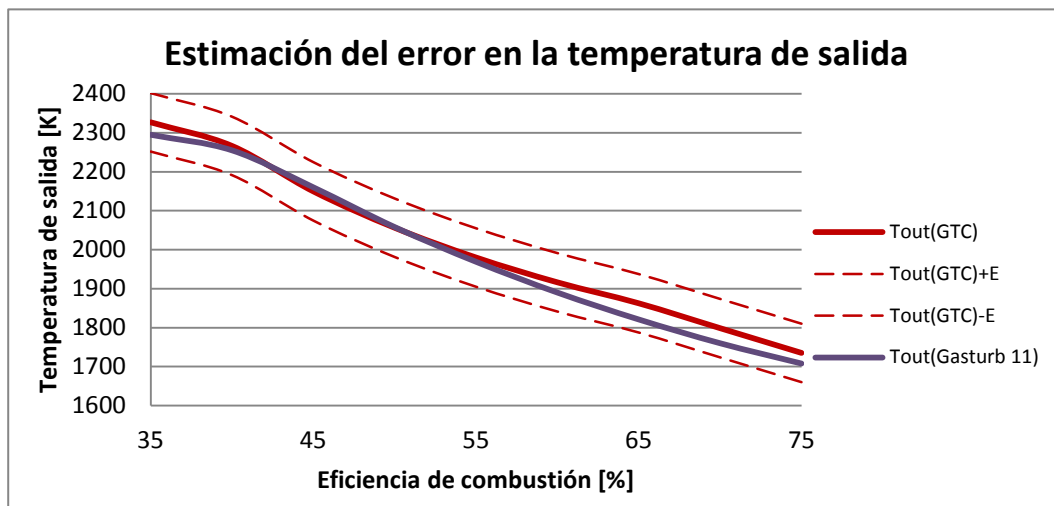


Fig. 5.10 Estimación del error y comparación de la temperatura de salida Tout(GTC) (obtenida en nuestro módulo con un valor de error +/-E) y la temperatura Tout(*GasTurb 11*) (obtenida por el programa comercial)

Podemos observar que los resultados de los módulos van parejos con los del programa comercial *GasTurb 11* (ver figura 5.10) con un error en la diferencia máxima de 41.49 K para una eficiencia del 65%. Concluimos así, que hemos logrado implementar un sistema de postcombustión suficientemente bueno pese a los posibles errores que se podrían haber introducido debido a la aproximación de las curvas.

5.4. Test de prueba del intercambiador de calor

En ese apartado se analizarán los diferentes resultados del módulo de grado máximo para el intercambiador de calor y se estimarán los posibles errores derivados de las ecuaciones.

5.4.1. Análisis de resultados obtenidos en el intercambiador

El intercambiador de calor de grado 2, como ya hemos visto, utiliza las mismas ecuaciones que el sistema de postcombustión para estimar las pérdidas de presión debidas a una inserción de calor en un conducto de área constante. Por esta razón, los resultados en este aspecto son equivalentes a los vistos en la sección 5.3.1. En cuanto al cálculo de la variación de temperaturas en el intercambiador hemos realizado una diferenciación entre dispositivos que trabajan a contracorriente y a corrientes paralelas.

Los resultados en dispositivos de corrientes paralelas para unas condiciones iniciales de temperatura de la zona caliente de 900K, una temperatura en la zona fría de 300 K, un flujo másico tanto de la zona caliente como de la zona fría de 20 Kg/s y un coeficiente global de intercambio de 0.6 KJ/m²Ks (en este caso hemos optado por un coeficiente de transferencia global típico para intercambiadores con agua) son:

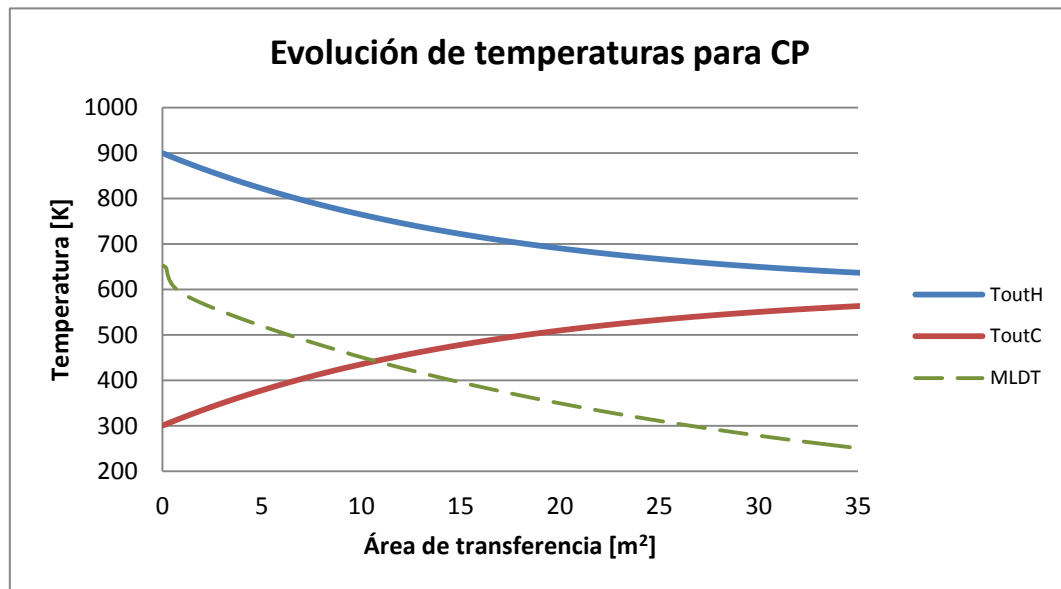


Fig. 5.11 Evaluación de temperaturas del lado caliente (ToutH), el lado frío (ToutC) y la temperatura media logarítmica (MLDT) en función del área de transferencia para una disposición a corrientes paralelas (CP)

En la figura 5.11 observamos como las corrientes no acaban de llegar al punto intermedio de 600 K tal y como se refleja en la figura 2.9. Teóricamente, para este tipo de disposiciones, las corrientes alcanzarán el valor medio para un área de transferencia infinita.

Los resultados de la disposición a contracorriente son algo distintos a los de la disposición a corrientes paralelas (utilizaremos las mismas condiciones iniciales que en la otra disposición). En primer lugar, las temperaturas de salida del lado frío y caliente para una determinada área de transferencia, pueden llegar a alcanzar las temperaturas del lado caliente y frío respectivamente. Podemos

apreciar este comportamiento en la figura 5.12 y como se corresponde con la de la figura 2.8.

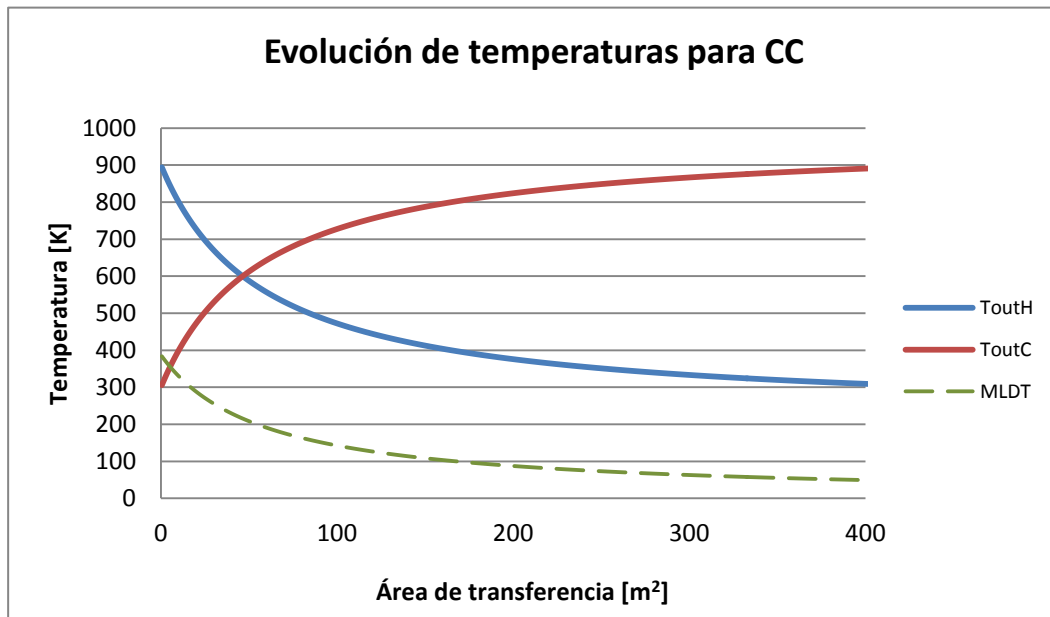


Fig. 5.12 Evaluación de temperaturas del lado caliente (Touth), el lado frío (ToutC) y la temperatura media logarítmica (MLDT) en función del área de transferencia para una disposición a contracorriente (CC)

De entre las dos disposiciones, se han realizado unas gráficas comparativas para comprobar cuál de las dos es la más efectiva (ver figura 5.13).

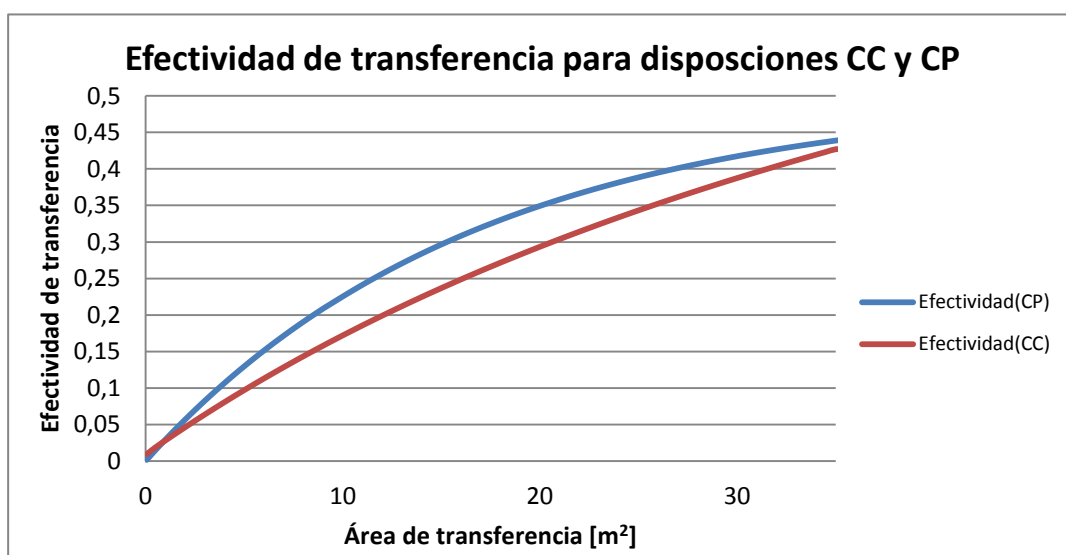


Fig. 5.13 Comparación entre la efectividad de transferencia para corrientes paralelas (CP) y contracorrientes (CC)

A primera vista, la disposición en corrientes paralelas parece poseer una mayor efectividad de transferencia. No obstante para obtener valores de efectividad mayores de 0.5, como por ejemplo en el caso 2 explicado en el apartado 2.3.1 de este documento, tendríamos que hacer uso de la disposición a contracorriente.

5.5. Test de prueba del motor ensamblado tipo 0

Junto con la colaboración de dos de mis compañeros encargados, por una parte del diseño de componentes relacionados con conductos y por otra la del diseño de turbo maquinaria, gracias a un esfuerzo conjunto hemos conseguido implementar el primer motor pre ensamblado turborreactor de grado 0. En este motor se han conseguido implementar con éxito los componentes básicos del turborreactor visto en la sección 1.2.2. Además, se han introducido los módulos de la atmosfera estándar, para evaluar las actuaciones del motor a diferentes altitudes, y el de parámetros del motor, para un análisis de sus características principales.

Para unas condiciones iniciales de 1000 m de altitud, una velocidad de entrada de gases de 100 m/s, un flujo másico de combustible de 4.7 kg/s, un flujo másico de entrada de aire de 235 kg/s, un poder calorífico de nuestro combustible de 43.1 MJ/kg y una temperatura de turbina fijada a 1250 K obtenemos:

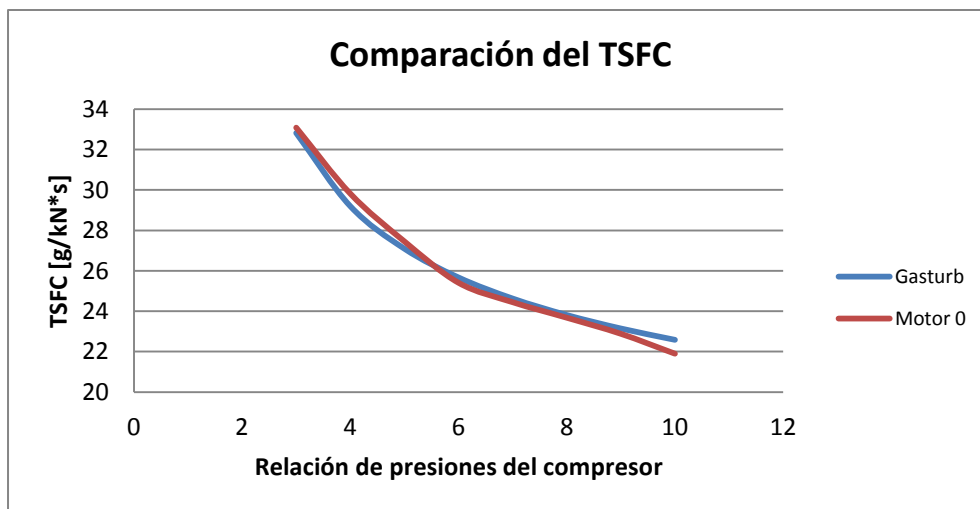


Fig. 5.14 Comparación del consumo específico de combustible por unidad de empuje (TSFC) entre nuestro programa de modelización 0 y el *GasTurb 11*

Observamos en la figura 5.14 como existe un error mínimo respecto a los resultados del programa comercial en el cálculo del TSFC. Se ha hecho la elección de este parámetro para la comparación del bloque motor completo porque es el que relaciona los valores característicos (empuje característico y

relación de combustible/aire) del motor turborreactor. Tenemos que señalar, que al hacer las pruebas en el *GasTurb 11* se han modificado los parámetros de entrada para que se corresponda con los de un motor turborreactor ideal.

Concluimos así, que los resultados de nuestro motor de grado 0 son suficientemente buenos en comparación con los del programa comercial y por lo tanto representa una buena referencia de la que partir.

5.6. Test de un estatorreactor

Junto con la colaboración del encargado en diseñar los módulos de conductos hemos creado con éxito un estatorreactor. En este caso ha sido necesaria la implementación de los modelos de grado 2 del difusor supersónico-subsonico, el sistema de postcombustión y la tobera convergente divergente. Tenemos en total, una serie de 46 ecuaciones con 74 conexiones entre los parámetros de las distintas funciones.

Para unas condiciones iniciales de Mach de entrada igual a 3, presión estática a la entrada y a la salida de 41 kPa, una temperatura estática a la entrada de 242.6 K (simulando las condiciones a 7000 m), un ángulo del núcleo central del difusor de 5 grados, una eficiencia del difusor de 0.95, una eficiencia de combustión de 0.36 y una relación de áreas en la tobera convergente divergente de 4 obtenemos:

Tabla 5.1. Resultados de las diferentes secciones del estatorreactor

Parámetros	Estatorreactor grado 2	Gasturb 11	Error relativo [%]
T03 [K]	679,42	670,15	1,36
T04 [K]	2215,83	2215,00	0,37
T05 [K]	2215,83	2159,45	2,54
P01 [kPa]	1498,03	1498,03	0
P02 [kPa]	608,68	1221,33	101,46
P03 [kPa]	604,48	1184,69	95,98
P04 [kPa]	593,11	1085,26	83,31
P05 [kPa]	593,11	1085,26	83,31
m3 [kg/s]	24,12	75,87	-
m4 [kg/s]	25,46	79,93	-
mf [kg/s]	1,34	4,05	-
M2	0,49	0,5	2,04
M3	0,10	0,1	0
M4	0,19	0,22	15,79
M5	2,94	2,72	7,48
FN [kN]	51,31	59,59	-
TSFC [g/kN*s]	26,11	68,05	-

La notación correspondiente a los valores de los parámetros mostrados en la tabla 5.1 se pueden observar en la figura 5.15.

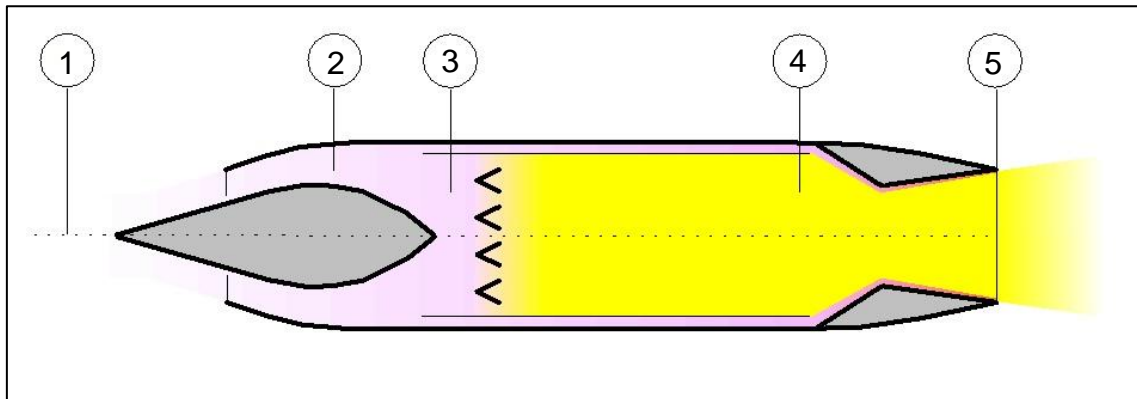


Fig. 5.15 Notación de los parámetros de la tabla 5.1

Observamos en la tabla 5.1 como los valores de temperatura totales (T_{02} , T_{03} , T_{04} y T_{05}) son prácticamente idénticos a los del programa comercial. En cambio, observamos que los valores de presiones totales se encuentran a la mitad. Este hecho se debe a que nuestro difusor supersónico-subsonico es programado para que simule una onda de choque oblicua y posteriormente una onda de choque normal. Esto hace que se produzca una caída de presión considerable y con ello una gran pérdida de eficiencia. Mientras que el *GasTurb 11* dispone de varias ondas de choque oblicuas que le proporcionan una caída de presión más moderada.

Además, también observamos una reducción de los valores de empuje (FN) y consumo específico de combustible debido al empuje (TSFC). Podemos deducir que se deben a la variación del valor de las diferentes áreas que componen la tobera y el difusor. Esto hace que se produzca una reducción de flujo másico y con ello las reducciones antes mencionadas.

En cuanto a la programación de este módulo, debemos señalar que la existencia de algunas funciones suficientemente complejas provocan que el margen de las estimaciones iniciales que podemos introducir sea reducido. Esto, sin duda limita el margen de operación del estatorreactor. La idea para futuros proyectos sería la de estudiar la inserción de valores adecuados para las estimaciones iniciales y así poder trabajar con un estatorreactor que funcione para un margen de operación mayor.

CAPÍTULO 6. CONCLUSIONES

Los objetivos de este trabajo inicialmente consistían en recopilar, estructurar, implementar y testear una serie de módulos de transferencia de calor. Todo ello mediante un proceso de escalado que contemplaría desde el caso más ideal hasta un caso más complejo. Además de la creación de un método de resolución de ecuaciones no lineales Newton-Raphson para un lenguaje de programación de alto nivel C++ para el testeo de los módulos en cuestión y el análisis de sus resultados.

Podemos concluir con éxito que se han llevado a cabo todos estos objetivos de forma satisfactoria además del montaje de dos tipos de motores previamente ensamblados (turbojet y ramjet). Esto sin duda refleja la dedicación y el empeño puesto en este proyecto y su gratificante recompensa. Aun así, ha significado toda una hazaña tanto de forma individual por el diseño y el testeo de los módulos de transferencia de calor, como globalmente a la hora de coordinarse con mis dos compañeros encargados de realizar otro tipo de módulos de la turbina de gas para el ensamblaje y funcionamiento de los motores.

Las dificultades no han sido pocas. Las principales están relacionadas con la búsqueda de información para grados de modelización elevados en los componentes de la cámara de combustión y postcombustión. Esto es debido a que en la implementación de este tipo de dispositivos se pasa de tener la ecuación más ideal a prácticamente necesitar la utilización de CFD. Como consecuencia se han empleado ecuaciones extraídas a partir de relaciones empíricas como grado de modelización máximo. Además, por parte del ámbito de programación desde la inserción de las librerías CLAPACK hasta la implementación de un sistema de resolución de Newton para sistemas de ecuaciones ha supuesto todo un reto para nuestro ingenio. Asimismo hemos comprobado que para grandes estructuras de funciones con ecuaciones muy complejas la determinación correcta de las estimaciones iniciales marcan la diferencia entre encontrar la solución o encontrar una errónea o incluso no alcanzar la convergencia.

Como futuros propósitos podemos decir que existen un gran número de frentes abiertos en cuanto al desarrollo y la implementación de un posible programa de simulación de motor de turbina modular. Desde un estudio de la inserción de las estimaciones iniciales que se deberían introducir en las funciones, hasta la mejora de cada uno de los módulos, pudiendo incluir técnicas avanzadas de CFD con mallas y aumentando el número de dimensiones. Además, se podría trabajar en una interface para el usuario donde este sea capaz de ensamblar su propio motor componente a componente de forma gráfica e intuitiva.

En definitiva, este trabajo ha significado mucho para nosotros y creemos firmemente en que pueda ser el primer paso hacia un software de renombre en el test de motores de turbina.

BIBLIOGRAFÍA

- [1] Lefebvre, A.H., *Gas Turbine Combustion*, third Edition, Taylor&Francis, New York, 2010.
- [2] Cohen, H., Rogers, G.F.C., Saravanamutto, H.I.H, *Gas Turbine Therory*, Longman Group Limited, England, 1996.
- [3] Hoffman, J.D., *Numerical Methods for Engineers and Scientist*, second edition, Marcel Dekker, New York, 2001.
- [4] Kern, D.Q., *Process Heat Transfer*, McGraw Hill, 1965.
- [5] Charles, E. O., *Aircraft Gas Turbine Power Plants*, Jeppesen, Colorado, 2010.
- [6] Esteban Oñate, A., *Turborreactores: Teoría, sistemas y propulsion de aviones*, Paraninfo, Madrid, 1981.
- [7] Cuesta Álvarez, M., *Motores de reacción: Tecnología y operación de vuelo*, Paraninfo, Madrid, 1995.
- [8] Cengel, Y., Boles, M., *Thermodynamics: An Engineering Approach*, McGraw-Hill, New York, 2010.
- [9] List, R. J., *Smithsonian Meteorological Tables*, Smithsonian Institution, Washintong, 1953.
- [10] Dugger, Gordon L., *Ramjets*, American Institute of Aeronautics and Astronautics, 1969.
- [11] Stull, R. B., *Meteorology for Scientists and Engineers*, Brooks/Cole, 2000.



Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

ANNEXOS

TÍTOL DEL TFC: Gas Turbine Design & Analysis Tool: Heat Transfer Components.

TITULACIÓ: Enginyeria Tècnica Aeronàutica, especialitat Aeronavegació

AUTOR: Jose Vigo Medina

DIRECTOR: Fernando Pablo Mellibovsky Elstein

DATA: 16 de juliol de 2012

ANEXOS

A1. Cálculo de la temperatura de transferencia de pared de cámara

Las ecuaciones utilizadas para el cálculo de la temperatura de transferencia de pared de la cámara de combustión sin refrigeración son:

$$R_1 = 0.5 * \sigma(1 + \epsilon_w)\epsilon_g T_g^{1.5}(T_g^{2.5} - T_w^{2.5})$$

$$C_1 = 0.020 \frac{k_g}{D_L^{0.2}} \left(\frac{\dot{m}_g}{A_L \mu_g} \right)^{0.8} (T_g - T_{w1})$$

O en la zona primaria:

$$C_{1,pz} = 0.017 \frac{k_g}{D_L^{0.2}} \left(\frac{\dot{m}_{pz}}{A_L \mu_g} \right)^{0.8} (T_g - T_{w1})$$

$$R_2 = Z\sigma(T_{w2}^4 - T_3^4)$$

Donde Z dependerá del material de la cubierta de la cámara. Para una cubierta de aluminio y acero estos valores serán de 0.4 y 0.6 respectivamente.

$$C_2 = 0.020 \frac{k_g}{D_{an}^{0.2}} \left(\frac{\dot{m}_{gan}}{A_{an} \mu_a} \right)^{0.8} (T_{w2} - T_3)$$

Y la ecuación que define el equilibrio:

$$R_1 + C_1 = R_2 + C_2 = \frac{k_w}{t_w} (T_{w1} - T_{w2}) = K_{1-2}$$

Como ecuaciones complementarias tenemos:

$$\epsilon_g = 1 - \exp[-290PL(qlb)^{0.5}T_g^{-1.5}]$$

$$L = 3(C/H - 5.2)^{0.75}$$

Donde R_1 es la radiación del flujo de calor desde la llama hasta el *liner* (Consideramos al *liner* como la pared de la cámara de combustión que separa la propia llama con el flujo de aire constante que se desvía inicialmente para refrigerar la propia cámara) , σ es la constante de Stefan-Boltzmann ($5.67 * 10^{-8} \frac{W}{m^2} * K^4$), ϵ_w es la emisividad de la pared del *liner*, ϵ_g es la emisividad del gas, T_g es la temperatura absoluta de gas, T_w es la temperatura absoluta de la pared del *liner*, C_1 es la convección del flujo de calor desde el *liner* hasta el aire que pasa por el “annulus”, k_g es la conductividad térmica del gas, D_L es el diámetro del *liner*, \dot{m}_g es el flujo másico del gas, A_L es el área transversal del *liner*, μ_g es la viscosidad dinámica del gas, R_2 es la radiación del flujo de calor desde el *liner* hasta la cubierta, K_{1-2} es la conducción del flujo de calor a lo largo de la pared del *liner*, P es la presión total, q es la relación de combustible/aire, lb es la longitud media del haz de trayectoria de la radiación, L es el factor de luminosidad y C/H es la relación entre el carbón/hidrógeno del combustible.

Para mayor detalle de la utilización y cálculo de estas ecuaciones consultar la referencia [1].

A2. Tabla de valores del coeficiente global de temperatura

COEFICIENTES TÍPICOS GLOBALES DE INTERCAMBIO “U”

CORRIENTE CALIDA	CORRIENTE FRÍA	Btu/(pie ² °F hr)	Kcal/(m ² °C hr)
Agua	Agua	140 - 280	86 - 1400
Solventes orgánicos	Agua	45 - 130	215 - 645
Gases	Agua	2.6 - 45	13 - 215
Aceites Livianos	Agua	60 - 160	300 - 770
Aceites Pesados	Agua	10 - 45	50 - 215
Solventes orgánicos	Aceites Livianos	20 - 70	100 - 345
Agua	Salmuera	105 - 210	515 - 1030
Solventes orgánicos	Salmuera	26 - 90	130 - 430
Gases	Salmuera	2.6 - 45	13 - 215
Solventes orgánicos	Solventes org.	20 - 62	100 - 300
Aceites Pesados	Aceites Pesados	8 - 44	40 - 215
Vapor	Agua	260 - 700	1290 - 3440
Vapor	Aceites Livianos	44 - 140	215 - 690
Vapor	Aceites Pesados	9 - 80	40 - 390
Vapor	Solventes org.	105 - 210	515 - 1030
Vapor	Gases	3.5 - 35	17 - 170
Fluidos de intercambio (tipo Dowtherm)			
	Aceites Pesados	8 - 53	38 - 260
Vapor Soluc. Acuosa	Baja viscosidad ($\mu < 2$ cP)	210 - 700	1030 - 3440
	Alta viscosidad ($\mu < 2$ cP)	105 - 210	515 - 1030

DATOS DE COEFICIENTE GLOBAL “U” PARA INDUSTRIA DE DESTILACION DE PETROLEO E INDUSTRIA PETROQUIMICA

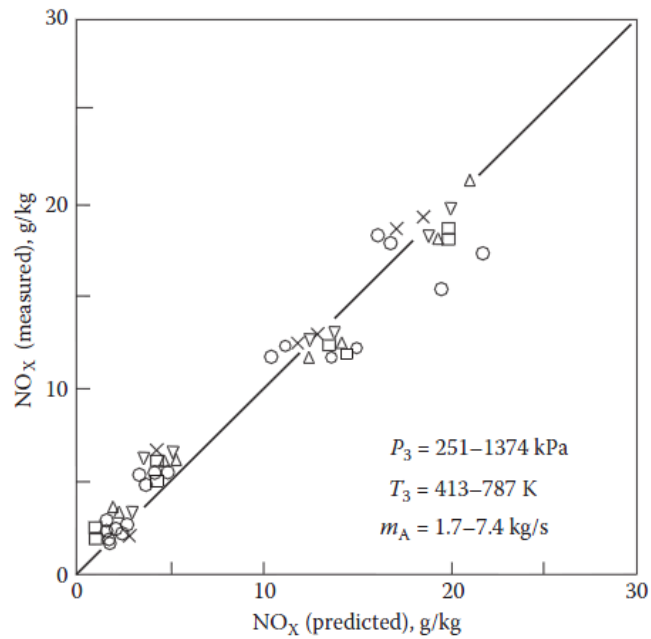
Fluido CALIDO	Fluido FRIO	U (BTU/Hora/pie ² /°F)
<i>Intercambiadores de Haz de Tubos y Coraza</i>		
Atmospheric Pipe-still Top Pumparound	Crude	60-70
Atmospheric P-s No. 3 Side streams	Crude	48-58
Atmospheric P-s Bottom Pumparound	Crude	55-85
Lean Oil	Fat Oil	60
Hydrocracker Effluent	Hydrocracker Feed	75
Hydrogenation Reactor Effluent	Hydrog.Reactor Feed	51-55
Hydrofiner Effluent	Hydrofiner Feed	50-68
Debutanizer Effluent	Debutanizer Feed	70
Powerformer Effluent	Powerformer Feed	50-80
Acetylene Converter Feed	Acety.Conv.Effluent	22-30
Regenerated D.E.A	Foul D.E.A.	110
Catalyst-Oil Slurry	Gas Oil Feed	40
Cracking Coil Vapors	Gas Oil	30
Rerun Still Overhead	Rerun Still Feed	50
Splitter Overhead	Debutanizer Feed	55
<i>Enfriadores</i>		
Brine	Water	150-210
Brine	Sour Water	100-115
Debutanizer Bottoms	Water	60-75
Debutanizer Overhead Products	Water	85-90
Depentanizer Bottom Products	Water	43
Vacuum Pipe Still Bottoms	Water	20-25
Absorber Oil	Water	80
Splitter Bottoms	Water	18
Lean Oil	Water	70
Heavy Gas Oil	Water	40
Regenerated D.E.A	Water	110
Reduced Crude	Water	29-32

<i>Enfriadores de Gas</i>			
Air 27 psig	Water		13
Air 105 psig	Water		17
Air 320 psig	Water		23
Primary Fractionator Gas	Water		27
Hydrocarbon Vapors (M.W = 30)	Water		38-43
Hydrocarbon Vapors (M.W = 25)	Water		55-60
Propylene	Water		50
Ethylene	Water		31
<i>Condensadores</i>			
Atmospheric Pipe Still Overhead	Water		80-90
Atmospheric Pipe Still Overhead	Crude		35-45
Atmospheric Pipe Still Distillate	Water		70-80
Vacuum Pipe Still Overhead	Water		115-130
Debutanizer Overhead	Water		90-100
Deethanizer Overhead	Water		90-113
Depentanizer Overhead	Water		110
Hydrofiner Effluent	Water		91-105
Stabilizer Overhead	Water		75-85
Splitter Overhead	Water		85-113
Rerun Still Overhead	Water		70
D.E.A. Regenerator Overhead	Water		100
Primary Fractionator Overhead	Water	40	(50% cond.)
Primary Fractionator Overhead and Products	Water	60	(25% cond.)
Powerformer Effluent	Water		55-60
Hydrocracker Effluent	Water		85
Propylene	Water		120
Steam	Water		400-600
<i>Congeladores</i>			
Ethylene	Propylene		98
Demethanizer Overhead	Ethylene		107
Deethanizer Overhead	Propylene		113
Depropanizer Overhead	Propylene		115
Ethylene	Ethylene		99-105
Demethanizer Feed	Ethylene		96-113
Demethanizer Feed	Propylene		100-122
<i>Rehervidores (calderetas)</i>			
Steam	Demethanizer Bottoms		75
Lean Oil	Demethanizer Bottoms		60
Steam	Deethanizer Bottoms		73-86
Atmospheric Pipe Still Top Pumparound	Deethanizer Bottoms		66
Steam	Depropanizer Bottoms		89
Steam	Debutanizer Bottoms		74-100
Atmospheric Pipe Still Top Pumparound	Debutanizer Bottoms		65
Atmospheric Pipe Still Bottoms	Debutanizer Bottoms		56
Steam	Depentanizer Bottoms		81
Steam	Debenzenizer Bottoms		102
Steam	Detoluenizer Bottoms		77
Steam	Splitter Bottoms		80
Dowtherm	Splitter Bottoms		70
Steam	Stripper Bottoms		82
Steam	Stabilizer Bottoms		115
Steam	Rerun Tower Bottoms		74
Dowtherm	Rerun Tower Bottoms		47
Steam	LPG Bottoms		70
Powerformer Effluent	Powerformer Stabilizer Bottoms		75-77
Steam	K3P04 Stripper Bottoms		145
Steam	D.E.A. Regenerator Bottoms		240
Dowtherm	Phenol		65

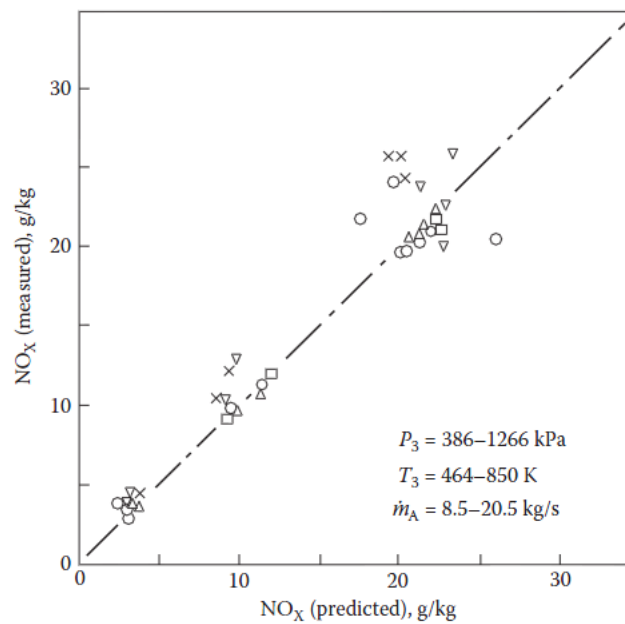
A3. Gráficas de emisiones

Estimacion de las emisiones de NO_x .

En el primer caso tenemos la gráfica que relaciona las emisiones de NO_x medidas y predecidas para un combustor J79-17A.

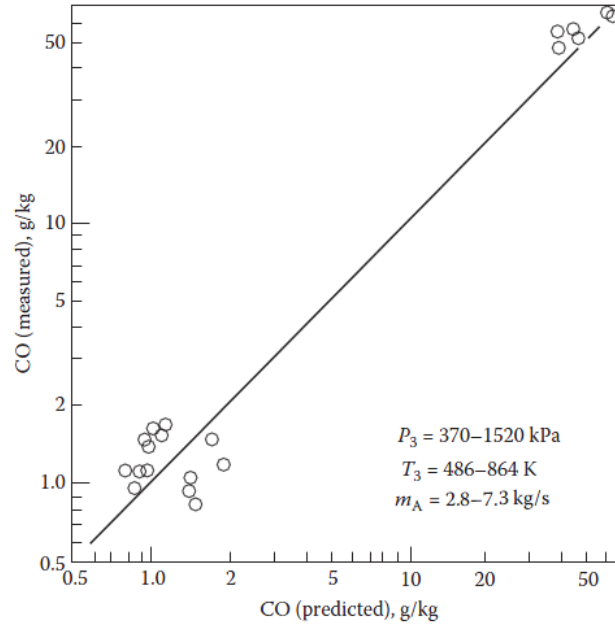


En el segundo caso tenemos la relación de emisiones de NO_x medidas y predecidas para un combustor GE F101.

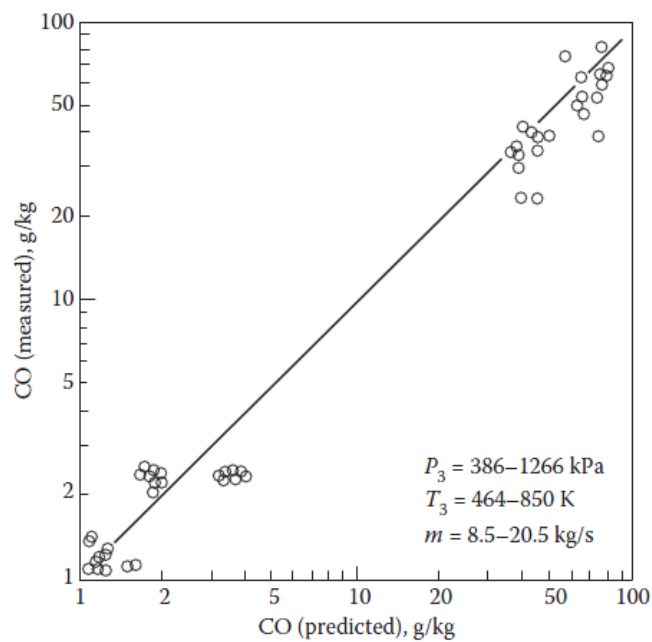


Estimación de las emisiones de CO.

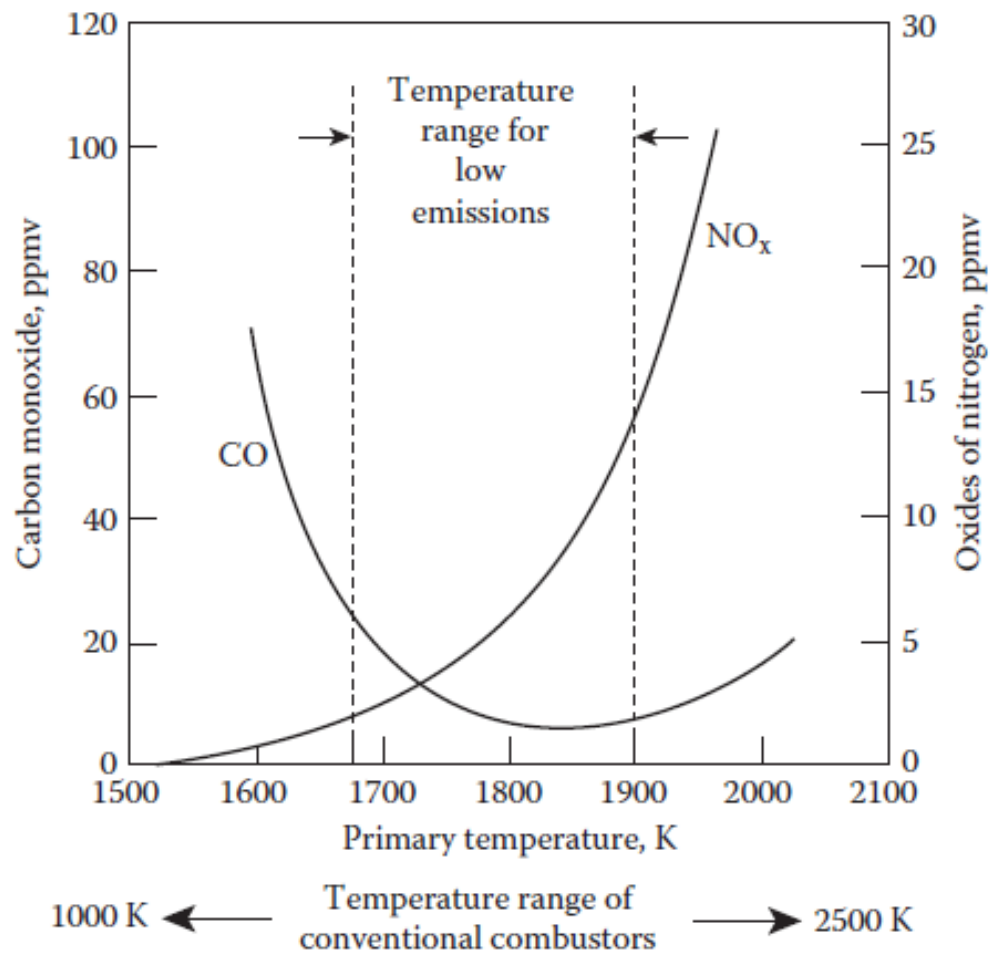
En el primer caso tenemos la gráfica que relaciona las emisiones de CO medidas y predecidas para un combustor P&W F100.



En el segundo caso tenemos la relación de emisiones de CO medidas y predecidas para un combustor GE F101.



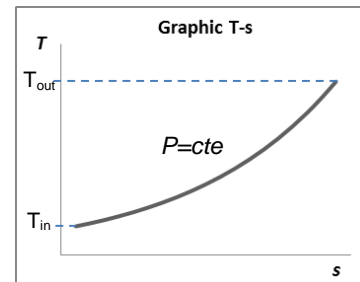
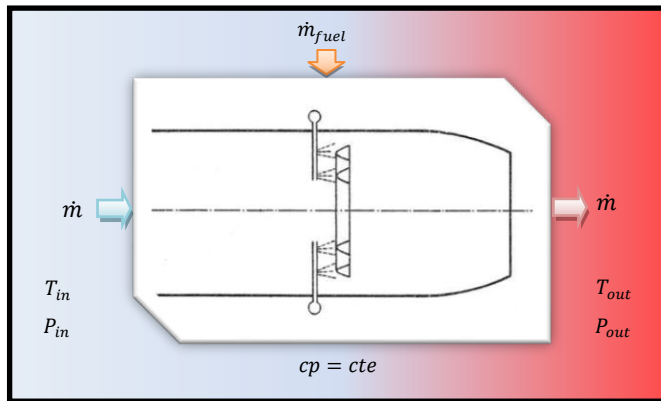
Influencia de la temperatura en la zona primaria en las emisiones de CO y NO_x.



A4. Lista de fichas de programación

Component: Afterburner Degree of modeling: 0

Images and graphics



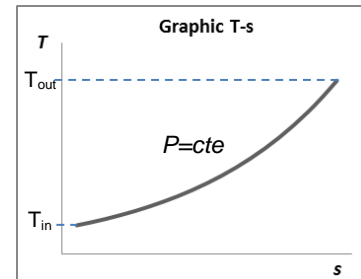
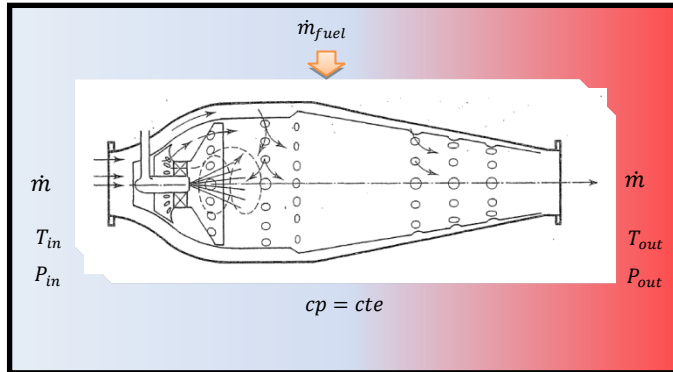
Description and assumptions

This device involves the insertion of heat through Q_{in} get increase our fluid energy.

- The addition of heat is produced at constant pressure.
- Ideal combustion conditions.
- The fluid is a perfect gas with constant specific heats and invariable composition
- The combustion chamber exploits all the specific heat of fuel.

Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Pressure, P. • Enthalpy, h. • Entropy, S. • Heat input, \dot{Q}_{in}. • Air mass flow rate, \dot{m}. • Fuel mass flow rate, \dot{m}_{fuel}. • Outlet pressure, P_{out}. • Inlet pressure, P_{in}. • Specifics heats, C_p & C_v. • Fuel heat power, q_{fuel}. 	Main
	$\dot{Q}_{in} = (h_{out} - h_{in}) = \dot{m} * cp(T_{out} - T_{in})$ $\dot{Q}_{in} = \dot{m}_{fuel} * q_{fuel}$ $P_{in} = P_{out}$
	Secondary
	$C_p = C_v + R$ $P_{in} = P_{out}$

Observations programming

Component: Combustion chamber**Degree of modeling: 0****Images and graphics****Description and assumptions**

This device involves the insertion of heat through Q_{in} to get increase our fluid energy.

- The addition of heat is produced at constant pressure.
- Ideal combustion conditions.
- The fluid is a perfect gas with constant specific heats and invariable composition
- The combustion chamber exploits all the specific heat of fuel.

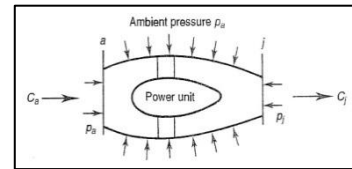
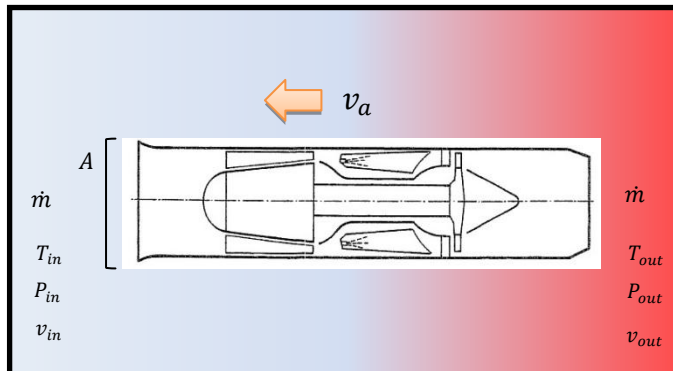
Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Pressure, P. • Enthalpy, h. • Entropy, S. • Heat input, \dot{Q}_{in}. • Air mass flow rate, \dot{m}. • Fuel mass flow rate, \dot{m}_{fuel}. • Outlet pressure, P_{out}. • Inlet pressure, P_{in}. • Specifics heats, C_p & C_v. • Fuel heat power, q_{fuel}. 	Main
	$E_{in} = E_{out}$ $\dot{Q}_{in} = (h_{out} - h_{in}) = \dot{m} * cp(T_{out} - T_{in})$ $\dot{Q}_{in} = \dot{m}_{fuel} * q_{fuel}$ $P_{in} = P_{out}$
	Secondary
	$C_p = C_v + R$

Observations programming

Component: Engine parameters

Degree of modeling: 0

Images and graphics



Description and assumptions

This module is responsible of the calculation of all parameters related to our gas turbine engine.

Variables

- Thrust Specific fuel consumption, **TSFC**
- Power Specific fuel consumption, **PSFC**
- Shaft Work, **W_{shaft}**
- Specific Thrust, **FS**
- Fuel/air ratio, **f**
- Cross sectional area, **A**
- Propulsion power, **\dot{W}_p**
- Propulsion efficiency, **η_p**
- Energy conversion efficiency, **η_e**
- Overall efficiency, **η_o**
- Input heat energy, **\dot{Q}_{in}**
- Velocity of the sistem, **v_a**
- Temperature, **T** .
- Pressure, **P** .
- Air mass flow rate, **\dot{m}** .
- Specifics heats, **C_p & C_v** .
- Fuel heat power, **q_{fuel}** .

Formulas

Main

$$F = \dot{m}(v_{out} - v_{in}) + (A(P_{out} - P_{in}))$$

$$\dot{W}_p = F * v_a$$

$$\dot{Q}_{in} = (f * q_{fuel}) * \dot{m}$$

$$\eta_p = \frac{\dot{W}_p}{\dot{Q}_{in}} = \frac{2}{1 + (v_{out}/v_{in})}$$

$$\eta_e = \frac{\dot{m}(v_{out}^2 - v_{in}^2)/2}{\dot{Q}_{in}}$$

$$\eta_o = \eta_e * \eta_p$$

$$FS = F/\dot{m}$$

$$TSFC = \frac{f}{FS} * 10^6 [g/KN * s]$$

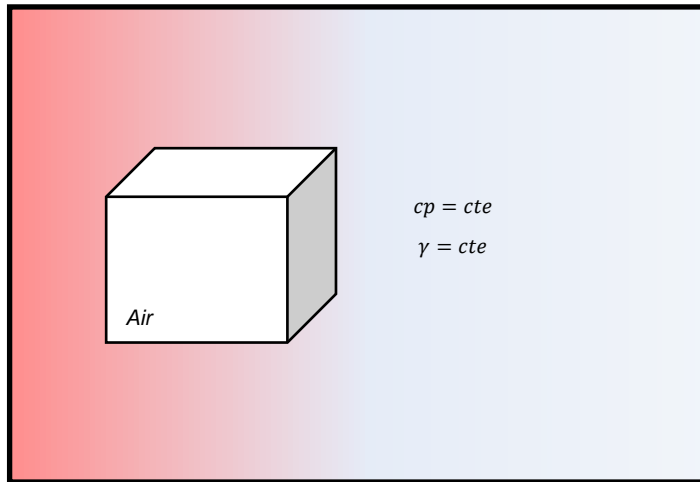
$$PSFC = \frac{f}{W_{shaft}} [Kg/Kw * s]$$

Secondary

Observations programming

Component: Flow characteristics

Degree of modeling: 0

Images and graphics**Description and assumptions**

This module is related to calculating the variations of the properties for the fluid and fluid type itself.

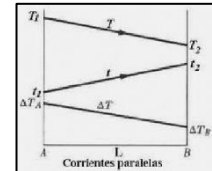
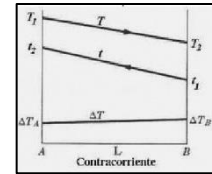
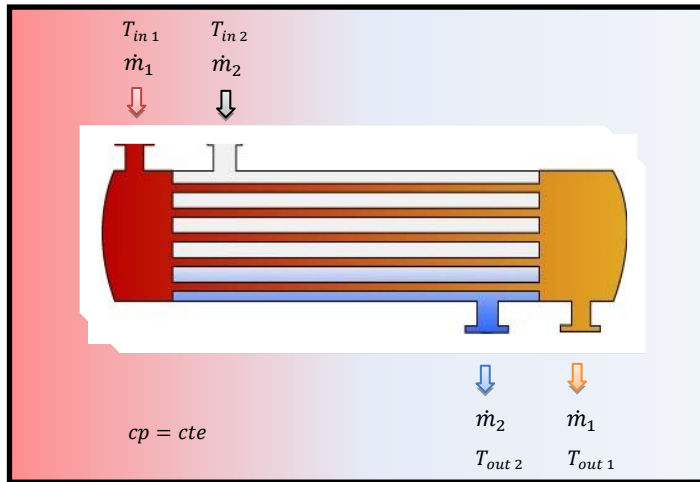
- The fluid is a perfect gas with constant specific heats and invariable composition
- We consider that the value of the constant specific heat of the air and the combustion gases are the same.
- The constant specific heats are independent of the fuel/air ratio of the combustion chamber and the effects of pressure.

Variables	Formulas
<ul style="list-style-type: none"> • Universal molar gas constant, \tilde{R}. • Ratio of specific heats, γ • Molecular weight, M • Specifics heats, C_p & C_v. 	Main
	$\frac{\gamma - 1}{\gamma} = \frac{\tilde{R}}{Mcp}$ $\frac{\tilde{R}}{M} = R \rightarrow R_{air} = 0.287 kJ/KgK$
	Secondary
	$C_p = 1.005 kJ/KgK$ $\gamma_a = 1.40$

Observations programming

Component: Heat exchanger Degree of modeling: 0

Images and graphics



Description and assumptions

This device involves the exchange of heat between two mass flows without mixing.

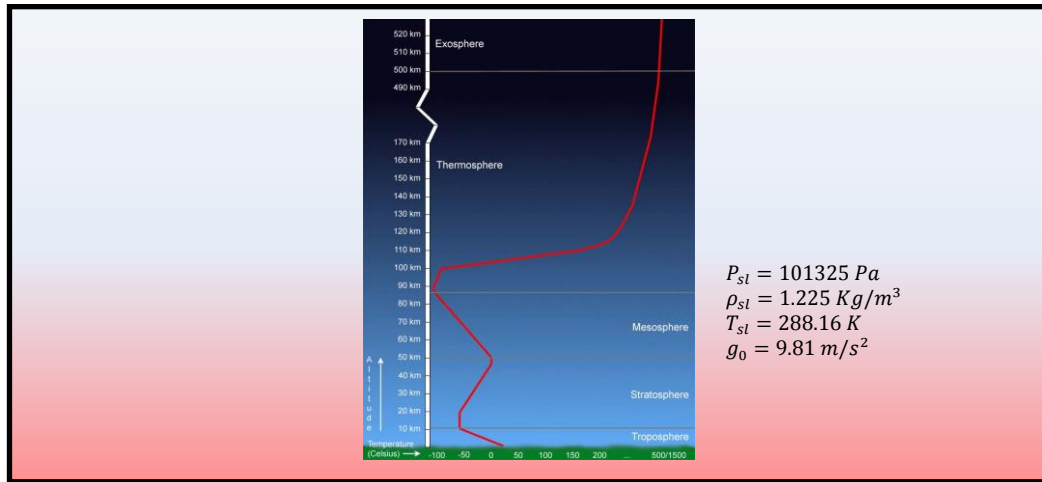
- The exchange of heat is produced in a stationary case.
- The fluids are a perfect gas with constants specific heats and invariable composition.
- We assume that the two heat-exchange fluids are equal and share the same specific heat.
- The fluids are working under adiabatic and isobaric conditions with the environment.

Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Enthalpy, h. • Entropy, S. • Air mass flow rate, \dot{m}. • Parallel streams, PS. • Streams countercurrent, SC. • Specifics heats, C_p & C_v. 	Main
	$\dot{m}_1(h_{1,in} - h_{1,out}) = \dot{m}_2(h_{2,out} - h_{2,in})$ $\Delta h = C_p(\Delta T)$ <p>Para PS: $T_{1,out} = T_{2,out}$</p> <p>Para SC: $T_{1,out} = T_{2,in}$ $T_{2,out} = T_{1,in}$</p>
	Secondary
	$C_p = C_v + R$

Observations programming

Component: Standard atmosphere Degree of modeling: 0

Images and graphics



Description and assumptions

This block simulates the conditions of temperature and pressure of a standard atmosphere.

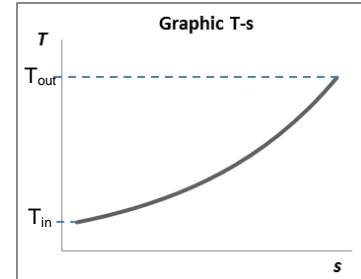
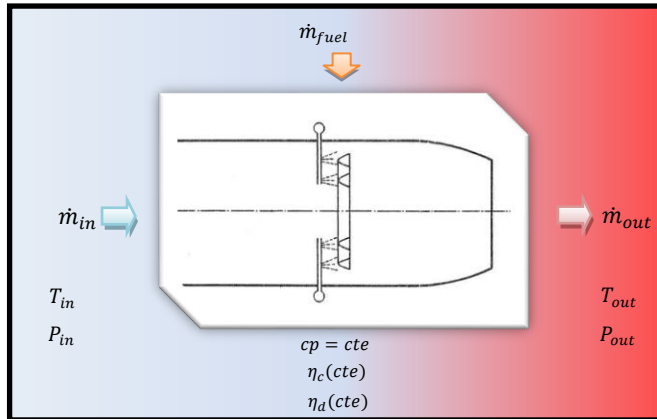
- The equation of state is referred to an ideal gas.
- The gases of the atmosphere are completely devoid of humidity and dust.
- The whole system is in hydrostatic equilibrium.

Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Pressure, P. • Density, ρ. • Altitude, h. • Gravity, g_0. • Gas constant air, R. • Temperature gradient, a_i. 	Main
	$\frac{P}{P_0} = \left(\frac{T}{T_0} \right)^{-\frac{g_0}{R \cdot a_i}}$ $\frac{P}{P_0} = e^{-\frac{g_0}{R \cdot T}(h-h_0)}$
	Secondary

Observations programming

Component: Afterburner

Degree of modeling: 1

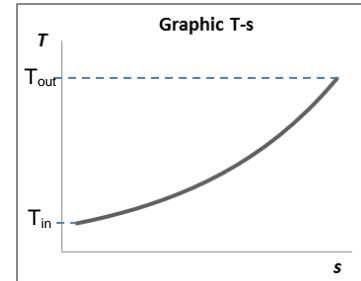
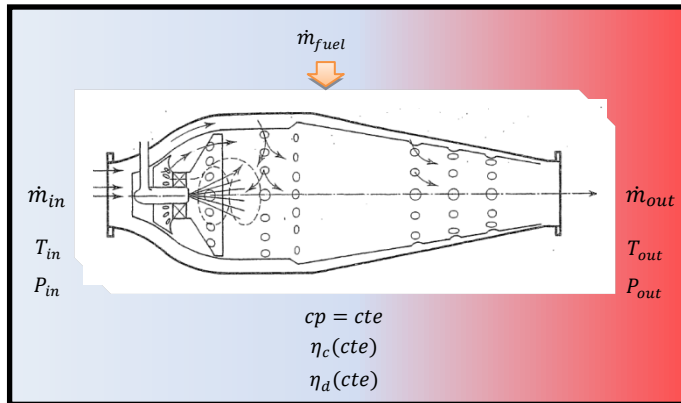
Images and graphics**Description and assumptions**

This device involves the insertion of heat through \dot{Q}_{in} get increase our fluid energy.

- The combustion efficiency and the momentum pressure loss are defined by a constants values (defined from typical values of an afterburner).
- Is considered an increase in mas flow due to the fuel insertions.
- The fluid is a perfect gas with constant specific heats and invariable composition.

Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Pressure, P. • Enthalpy, h. • Entropy, S. • Heat input, \dot{Q}_{in}. • Momentum pressure loss, η_d • Combustion efficiency, η_c • Input air mass flow rate, \dot{m}_{in}. • Output gases mass flow rate, \dot{m}_{out}. • Fuel mass flow rate, \dot{m}_{fuel}. • Specifics heats, C_p & C_v. • Fuel heat power, q_{fuel}. 	Main
	$\dot{Q}_{in} = (h_{out} - h_{in}) = \dot{m}_{out} * cp(T_{out} - T_{in})$ $\dot{Q}_{in} = \eta_c * \dot{m}_{fuel} * q_{fuel}$
	Secondary
	$C_p = C_v + R$ $\dot{m}_{out} = \dot{m}_{in} + \dot{m}_{fuel}$ $\eta_c = 0.4$ $\eta_d = \frac{P_{in} - P_{out}}{P_{in}} = 0.06$ $q_{fuel}(kerosene) = 43100 \text{ Kj/kg}$

Observations programming

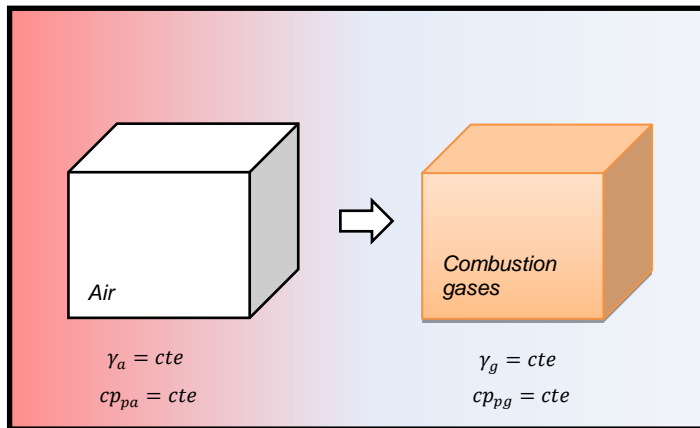
Component: Combustion chamber**Degree of modeling: 1****Images and graphics****Description and assumptions**

This device involves the insertion of heat through Q_{in} to get increase our fluid energy.

- The combustion efficiency and the momentum pressure loss are defined by a constants values (defined from typical values of combustion chamber).
- Is considered an increase in mas flow due to the fuel insertions.
- The fluid is a perfect gas with constant specific heats and invariable composition

Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Pressure, P. • Enthalpy, h. • Entropy, S. • Heat input, \dot{Q}_{in}. • Momentum pressure loss, η_d • Combustion efficiency, η_c • Input air mass flow rate, \dot{m}_{in}. • Output gases mass flow rate, \dot{m}_{out}. • Fuel mass flow rate, \dot{m}_{fuel}. • Specifics heats, C_p & C_v. • Fuel heat power, q_{fuel}. 	Main
	$\dot{Q}_{in} = (h_{out} - h_{in}) = \dot{m}_{out} * cp(T_{out} - T_{in})$ $\dot{Q}_{in} = \eta_c * \dot{m}_{fuel} * q_{fuel}$
	Secondary
	$C_p = C_v + R$ $\eta_d = \frac{P_{in} - P_{out}}{P_{in}} = 0.06$ $\eta_c = 0.95$ $\dot{m}_{out} = \dot{m}_{in} + \dot{m}_{fuel}$ $q_{fuel}(kerosene) = 43100 \text{ Kj/kg}$

Observations programming

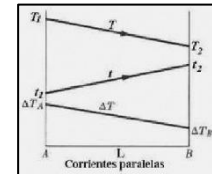
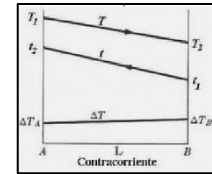
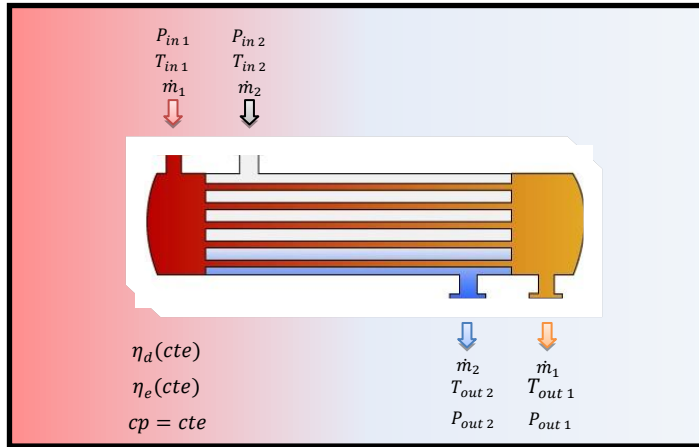
Component: Flow characteristics**Degree of modeling: 1****Images and graphics****Description and assumptions**

This module is related to calculating the variations of the properties for the fluid and fluid type itself.

- The fluid is a perfect gas with constant specific heats.
- We consider different values of the constant specific heat for the air and the combustion gases.
- The constant specific heats are independent of the fuel/air ratio of the combustion chamber and the effects of pressure.

Variables	Formulas
<ul style="list-style-type: none"> • Universal molar gas constant, \tilde{R}. • Ratio of specific heats(combustion gases), γ_g • Ratio of specific heats(air), γ_a • Molecular weight, M • Specific heat (combustion gases), C_{pg} • Specific heat (air), C_{pa}. 	Main
	$\frac{\gamma - 1}{\gamma} = \frac{\tilde{R}}{Mcp}$ $\frac{\tilde{R}}{M} = R$
	Secondary
	$C_{pa} = 1.005 \text{ Kj/KgK}$ $\gamma_a = 1.40$ $C_{pg} = 1.148 \text{ Kj/KgK}$ $\gamma_g = 1.333$

Observations programming

Component: Heat exchanger**Degree of modeling: 1****Images and graphics****Description and assumptions**

This device involves the exchange of heat between two mass flows without mixing.

- The exchange of heat is estimated with a constant value of effectiveness.
- The pressure loss is defined by constants values.
- The fluids are a perfect gas with constants specifics heats and invariable composition.

Variables

- Temperature, T
- Pressure, P
- Entropy, S
- Air mass flow rate at heat side, \dot{m}_1
- Air mass flow rate at cold side, \dot{m}_2
- Pressure loss (cold side), $\eta_{d,c}$
- Pressure loss (hot side), $\eta_{d,h}$
- Effectiveness of the heat exchanger, η_e
- Specifics heats, C_p & C_v

Formulas**Main**

$$\dot{m}_1(h_{1,in} - h_{1,out}) = \dot{m}_2(h_{2,out} - h_{2,in})$$

$$\Delta h = Cp(\Delta T)$$

$$\eta_e = 1 - \frac{T_{1,out} - T_{2,in}}{T_{1,in} - T_{2,in}}$$

$$\eta_{d,h} = \frac{P_{1,in} - P_{1,out}}{P_{1,in}}$$

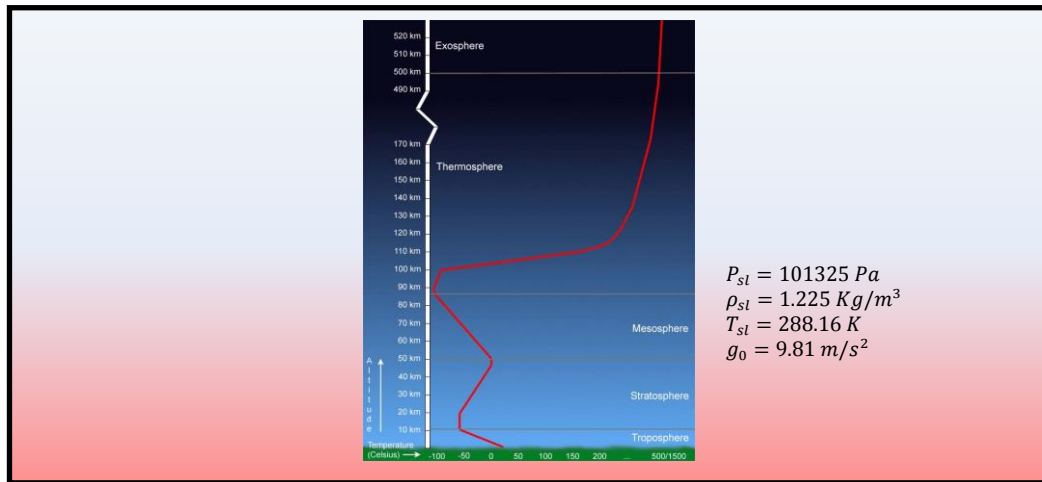
$$\eta_{d,c} = \frac{P_{2,in} - P_{2,out}}{P_{2,in}}$$

Secondary

$$C_p = C_v + R$$

$$\eta_e = 0.9$$

Observations programming

Component: Standard atmosphere**Degree of modeling: 1****Images and graphics****Description and assumptions**

This block simulates the conditions of temperature and pressure of a standard atmosphere.

- The equation of state is referred to an ideal gas.
- The gases of the atmosphere are completely devoid of humidity and dust.
- The whole system is in hydrostatic equilibrium.

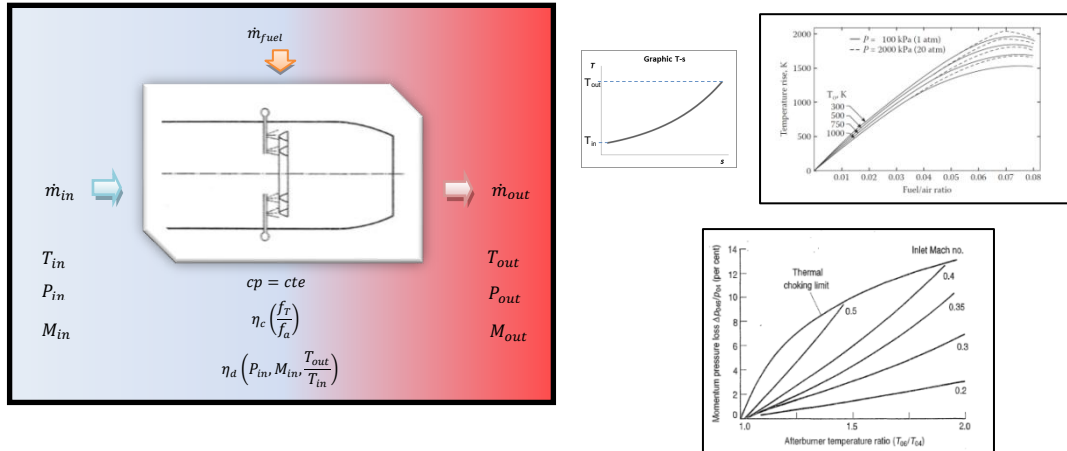
Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Static pressure, P_s. • Dynamic pressure, P_d. • Total pressure, P_t. • Speed of sound, v_s • Velocity, v. • Number of Mach, M. • Density, ρ. • Gravity, g_0. • Gas constant air, R. • Ratio of specific heats, γ • Temperature gradient, a_i. 	Main
	$\frac{P_s}{P_{s0}} = \left(\frac{T}{T_0} \right)^{-\frac{g_0}{R \cdot a_i}}$ $\frac{P_s}{P_{s0}} = e^{-\frac{g_0}{R \cdot T}(h-h_0)}$
	Secondary
	$v_s = \sqrt{\gamma \cdot R \cdot T}$ $M = v/v_s$ $P_d = \frac{\rho \cdot v^2}{2}$ $P_t = P_s + P_d$

Observations programming

Component: Afterburner

Degree of modeling: 2

Images and graphics



Description and assumptions

This device involves the insertion of heat through Q_{in} get increase our fluid energy.

- The addition of head is estimated in function of the fuel/air ratio, the input pressure and the input temperature from temperature-rise curves for kerosene-air mixtures.
- Is considered an increase in mass flow due to the fuel insertions.
- The combustion efficiency is estimated in function of the air/fuel ratio.
- The pressure loss is estimated in function of the inlet number of Mach and the afterburner temperature ratio.
- The fluid is a perfect gas with constant specific heats and invariable composition

Variables

- Temperature, T .
- Pressure, P .
- Entropy, S .
- Heat input, Q_{in} .
- Momentum pressure loss, η_d
- Combustion efficiency, η_c
- Fuel/air ratio, f
- Speed of sound, v_s
- Velocity, v
- Mach number, M
- Input air mass flow rate, \dot{m}_{in} .

Formulas

Main

$$\dot{Q}_{in} = (h_{out} - h_{in}) = \dot{m}_{out} * cp(\Delta T_{comb})$$

$$T_{out} = T_{in} + \Delta T_{comb}(f, T_{in}, P_{in})$$

$$\eta_c = \frac{f_{theoretical}}{f_{actual}}$$

$$\eta_d = 1 - \frac{P_{out}}{P_{in}}$$

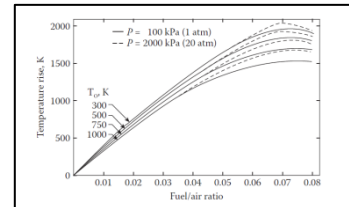
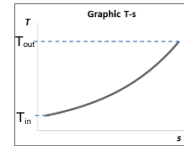
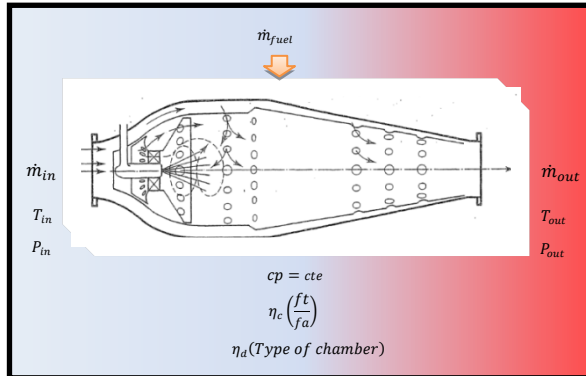
$$\frac{P_{out}}{P_{in}} = \frac{P_{out}}{P^*} * \frac{P^*}{P_{in}}$$

*We obtain ΔT_{comb} from the tables of the reference [1]

<ul style="list-style-type: none"> • Output gases mass flow rate, \dot{m}_{out}. • Fuel mass flow rate, \dot{m}_{fuel}. • Specifics heats, C_p & C_v. • Ratio of specific heats, γ • Fuel heat power, q_{fuel}. 	<p>Secondary</p> $C_p = C_v + R$ $\dot{m}_{out} = \dot{m}_{in} + \dot{m}_{fuel}$ $q_{fuel}(kerosene) = 43100 \text{ Kj/kg}$ $\frac{P_{in}}{P^*} = \left[\frac{1 + \gamma}{1 + \gamma M^2} \right] \left[\frac{2 + (\gamma - 1)M^2}{\gamma + 1} \right]^{\frac{\gamma}{\gamma - 1}}$ $\frac{T_{in}}{T^*} = \frac{(1 + \gamma)M^2}{(1 + \gamma M^2)^2} [2 + (\gamma - 1)M^2]$ $\frac{T_{out}}{T^*} = \left(\frac{T_{out}}{T_{in}} \right) \left(\frac{T_{in}}{T^*} \right)$ $v_s = \sqrt{\gamma R T}$ $M = \frac{v}{v_s}$
---	---

Observations programming

The maximum value for M_{out} must be 1 due to the thermal choking limit.

Component: Combustion chamber**Degree of modeling: 2****Images and graphics****Description and assumptions**

This device involves the insertion of heat through Q_{in} to get increase our fluid energy.

- The addition of head is estimated in function of the fuel/air ratio, the input pressure and the input temperature from temperature-rise curves for kerosene-air mixtures.
- The combustion efficiency is estimated in function of the fuel/air ratio.
- The momentum pressure loss is defined by a constants values depending of the type of chamber.
- Is considered an increase in mas flow due to the fuel insertions.
- The fluid is a perfect gas with constant specific heats and invariable composition

Variables

- Temperature, T .
- Pressure, P .
- Entropy, S .
- Heat input, Q_{in} .
- Fuel/air ratio, f
- Momentum pressure loss, η_d
- Combustion efficiency, η_c
- Input air mass flow rate, \dot{m}_{in} .
- Output gases mass flow rate, \dot{m}_{out} .
- Fuel mass flow rate of primary zone, \dot{m}_{pz}
- Specifics heats, C_p & C_v .
- Fuel heat power, q_{fuel} .

Formulas**Main**

$$\dot{Q}_{in} = (h_{out} - h_{in}) = \dot{m}_{out} * cp(\Delta T_{comb})$$

$$T_{out} = T_{in} + \Delta T_{comb}(f, T_{in}, P_{in})$$

*We obtain ΔT_{comb} from the tables of the reference [1]

Secondary

$$C_p = C_v + R$$

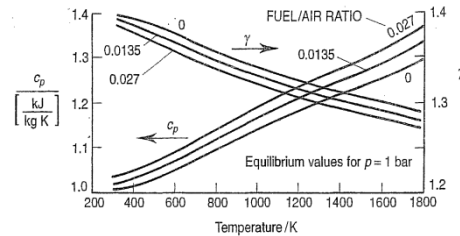
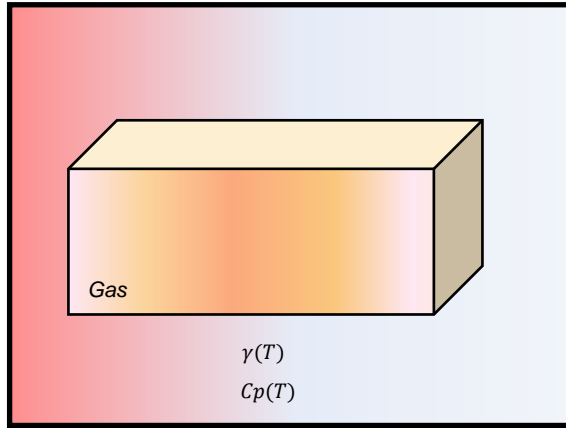
$$\eta_d(Tubular) = \frac{\Delta P}{P_{in}} = 0.07$$

$$\eta_d(Tuboannular) = 0.06$$

$$\eta_d(Annular) = 0.06$$

<ul style="list-style-type: none"> • Combustion volume, V_c • Stoichiometric flame temperature, T_{st} • Number of cylinders of the combustor (tubular or can-annular), n • Cross sectional area of one can, A_{ref} • Liner length, L • Volume employed in fuel evaporation, V_e • Density of the gas in the primary zone, ρ_{pz} • Evaporation constant, λ_{eff} • Sauter mean diameter, D_0 • Temperature at primary zone, T_{pz} • Unburned Hydrocarbons, <i>UHC</i> • Carbon Monoxide emissions, <i>CO</i> • Oxides of Nitrogen emissions, <i>NO_x</i> 	$\eta_c = \frac{f_{theoretical}}{f_{actual}}$ $\dot{m}_{out} = \dot{m}_{in} + \dot{m}_{fuel}$ <p>Emissions:</p> $NO_x = \frac{9 * 10^{-8} P_{in}^{1.25} V_c * e^{0.01 T_{st}}}{\dot{m}_{in} * T_{pz}}$ $V_c = n_{t,c} * L * A_{ref}$ $\eta_{c,pz} = 0.85$ $T_{pz} = T_{in} + \Delta T_{comb} - 55$ $CO = \frac{86 \dot{m}_{in} T_{pz} * e^{(-0.00345 T_{pz})}}{(V_c - V_e) \left(\frac{\Delta P}{P}\right)^{0.5} * P_{in}^{1.5}}$ $\dot{m}_{pz} = 0.26 * \dot{m}_{in}$ $V_e = 0.55 * \dot{m}_{pz} * D_0^2 / \rho_{pz} \lambda_{eff}$ $UHC = \frac{(1 - \eta_c)}{10^{-3}} - 0.232 * CO$
--	--

Observations programming

Component: Flow characteristics**Degree of modeling: 2****Images and graphics****Description and assumptions**

This module is related to calculating the variations of the properties for the fluid and fluid type itself.

- The fluid is a perfect gas with variable specific heat as a function of temperature.
- The constant specific heats are dependent of the fuel/air ratio of the combustion chamber, but we consider negligible the effects of pressure.

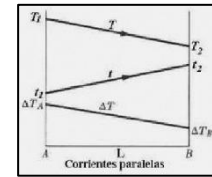
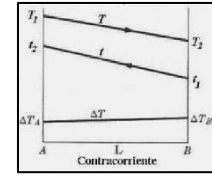
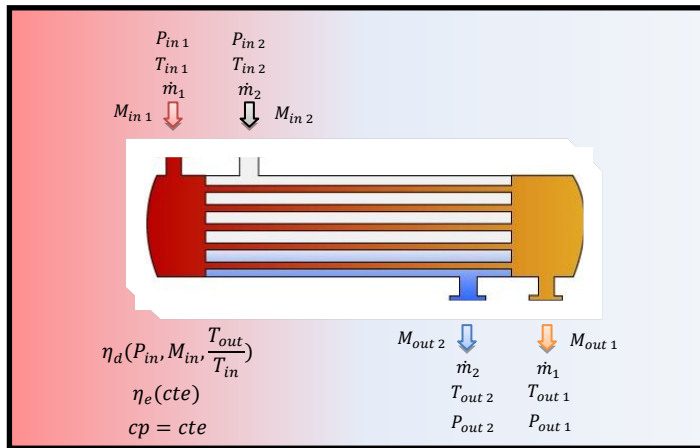
Variables	Formulas
<ul style="list-style-type: none"> • Universal molar gas constant, \tilde{R}. • Ratio of specific heats(combustion gases), γ_g • Ratio of specific heats(air), γ_a • Molecular weight, M • Specific heat (combustion gases), C_{pg} • Specific heat (air), C_{pa}. 	Main
	$\frac{\gamma - 1}{\gamma} = \frac{\tilde{R}}{Mcp}$ <p>*We interpolate in the figure 2.14 of the reference [2] to obtain the values of c_p and γ.</p>
	Secondary

Observations programming

Component: Heat exchanger

Degree of modeling: 2

Images and graphics



Description and assumptions

This device involves the exchange of heat between two mass flows without mixing.

- The exchange of heat is estimated with a constant value of effectiveness.
- The pressure loss is estimated in function of the inlet number of Mach and the heat exchanger temperature ratio of every duct.
- The fluids are a perfect gas with constants specific heats and invariable composition.

Variables

- Temperature, T
- Pressure, P
- Entropy, S
- Air mass flow rate at heat side, \dot{m}_1
- Air mass flow rate at cold side, \dot{m}_2
- Pressure loss (cold side), $\eta_{d,c}$
- Pressure loss (hot side), $\eta_{d,h}$
- Effectiveness of the heat exchanger, η_e
- Parallel streams, PS.
- Streams countercurrent, SC.
- Logarithmic mean

Formulas

Main

$$\dot{m}_1(h_{1,in} - h_{1,out}) = \dot{m}_2(h_{2,out} - h_{2,in})$$

$$\Delta h = Cp(\Delta T)$$

$$\eta_e = 1 - \frac{T_{2,out} - T_{2,in}}{T_{1,in} - T_{1,out}}$$

$$\eta_{d,h} = \frac{P_{2,in} - P_{2,out}}{P_{2,in}}$$

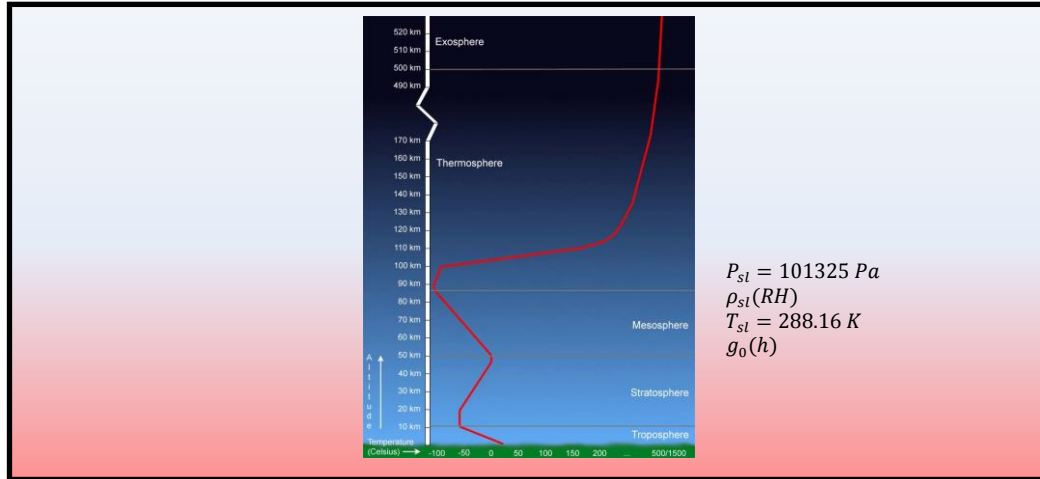
$$\eta_{d,c} = \frac{P_{2,in} - P_{2,out}}{P_{2,in}}$$

$$\left(\frac{P_{out}}{P_{in}}\right)_{h \text{ or } c} = \left(\frac{P_{out}}{P^*} * \frac{P^*}{P_{in}}\right)$$

Secondary

<p>temperature, MLDT.</p> <ul style="list-style-type: none"> • Transfer area, A. • Global transfer coefficient, U. • Ratio of specific heats, γ • Mach number, M • Velocity, v • Speed of sound, v_s • Specific heats, C_p & C_v 	$C_p = C_v + R$ $\eta_e = 0.9$ $\frac{P_{in}}{P^*} = \left[\frac{1 + \gamma}{1 + \gamma M^2} \right] \left[\frac{2 + (\gamma - 1)M^2}{\gamma + 1} \right]^{\frac{\gamma}{\gamma - 1}}$ $\frac{T_{in}}{T^*} = \frac{(1 + \gamma)M^2}{(1 + \gamma M^2)^2} [2 + (\gamma - 1)M^2]$ $\frac{T_{out}}{T^*} = \left(\frac{T_{out}}{T_{in}} \right) \left(\frac{T_{in}}{T^*} \right)$ $v_s = \sqrt{\gamma R T}$ $M = \frac{v}{v_s}$ $A = \frac{\dot{m}_1 c_{p1} (T_{1,in} - T_{1,out})}{U * \Delta T}$ $MLDT_{PS} = \frac{(T_{1,in} - T_{2,in}) - (T_{1,out} - T_{2,out})}{\ln \left(\frac{T_{1,in} - T_{2,in}}{T_{1,out} - T_{2,out}} \right)}$ $MLDT_{SC} = \frac{(T_{1,in} - T_{2,out}) - (T_{1,out} - T_{2,in})}{\ln \left(\frac{T_{1,in} - T_{2,out}}{T_{1,out} - T_{2,in}} \right)}$ $\Delta T = MLDT$
--	---

Observations programming

Component: Standard atmosphere**Degree of modeling: 2****Images and graphics****Description and assumptions**

This block simulates the conditions of temperature and pressure of a standard atmosphere.

- The equation of state is referred to an ideal gas.
- The gases of the atmosphere are completely devoid of humidity and dust.
- The whole system is in hydrostatic equilibrium.
- There are corrections of gravity in function of the altitude and the latitude.
- There are corrections for the density in function of the relative humidity.

Variables	Formulas
<ul style="list-style-type: none"> • Temperature, T. • Static pressure, P_s. • Dynamic pressure, P_d. • Total pressure, P_t. • Speed of sound, vs. • Velocity, v. • Number of Mach, M. • Density, ρ. • Gravity, g_0. • Gas constant air, R. • Ratio of specific heats, γ. • Relative humidity, RH [%] 	Main
	$\frac{P_s}{P_{s0}} = \left(\frac{T}{T_0}\right)^{-\frac{g_0}{R \cdot a_i}}$ $\frac{P_s}{P_{s0}} = e^{-\frac{g_0}{R \cdot T}(h-h_0)}$
	Secondary
	$vs = \sqrt{\gamma \cdot R \cdot T}$ $M = v/vs$

<ul style="list-style-type: none"> • Latitude, ϕ • Temperature gradient, a_i. • Universal gravitational constant, G • Mass of the earth, M_t • Angular velocity of the earth, ω • Radius of the earth, R_t. • Saturation pressure of water vapor, E_s. • Constants, C_0, C_1, C_2. • Pressure of water vapor, P_v. 	$Pd = \frac{\rho * v^2}{2}$ $Pt = Ps + Pd$ $g^* = G * \frac{M_t}{(R_t + h)^2}$ $ac = R_t * \cos(\phi) * \omega$ $g = \sqrt{(g^*)^2 + ac^2 - (2g^*ac \cos(\phi))}$ $E_s = C_0 * 10^{\frac{C_1 * T}{C_2 + T}}$ $P_v = RH * E_s$ $\rho = \left(\frac{P}{R * T} \right) * \left(1 - \frac{0.378 * P_v}{P} \right)$
--	--

Observations programming

A5. Código de programación (Newton-Raphson)

A continuación se mostrará el fichero .cpp en el que se ha implementado el método de Newton con el procedimiento “*Lagged Newton’s method*” para el módulo del sistema de postcombustión de grado 2.

```
#include "Combustor.h"
#include "Afterburner.h"
#include "Heat_exchanger.h"
#include "FlowC.h"
#include "Atmosphere_STD.h"
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "f2c.h"
#include "clapack.h"

#define MAX 30

void main()
{
    //Definicion de parametros
    double F1,F2,F3,delta_x=0.001;
    double Tin=800, Pin=450000, nc=1, vin=200, gamma=1.4, R=286.9, ft=0.02,
    mfuel=-1, min=28.8;
    double Tout=-1, fa=-1,mout=-1,nd=-1, Pr1=-1, Pr3=-1, Pr2=-1, Main=-1, Maout=-
    1, Tr1=-1, Tr2=-1, vout=-1;
    int n, nc1=14, r=0,c=0,a=0,b=0, cont=0;           //nc: numero de conexiones
    int i=0,j=0,z=0;
    double A[200],dFx[30];

    //Definicion de matrices del metodo de newton-raphson
    double guess=1,sol[MAX],F[MAX],dfx[200];
    int x[MAX][MAX][MAX],xn=0;

    //Se debe modificar al aumentar el número de ecuaciones
    //conexiones
    double conex[MAX][MAX][MAX];
    int links[MAX][MAX];

    i=0;
    j=0;
    z=0;
    while(i<=MAX)
    {
        j=0;
        while(j<3)
        {
            z=0;
            while(z<=MAX)
            {
                conex[i][j][z]=MAX-1;

                z++;
            }
            j++;
        }
        i++;
    }
    /*conex[1][0][0]=0; // ecuacion
```

```

conex[1][0][1]=1;    // Posicion en la ecuacion
*/
    //conexió Tout
conex[1][0][0]=0;
conex[1][0][1]=1;
conex[1][1][0]=9;
conex[1][1][1]=1;
conex[1][2][0]=11;
conex[1][2][1]=4;

    //conexió1 fa
conex[2][0][0]=0;
conex[2][0][1]=2;
conex[2][1][0]=1;
conex[2][1][1]=1;
conex[2][2][0]=12;
conex[2][2][1]=0;

    //conexió Tr2
conex[3][0][0]=2;
conex[3][0][1]=2;
conex[3][1][0]=12;
conex[3][1][1]=2;

    //conexió Pr1
conex[4][0][0]=3;
conex[4][0][1]=1;
conex[4][1][0]=4;
conex[4][1][1]=0;

    //conexió Pr2
conex[5][0][0]=4;
conex[5][0][1]=1;
conex[5][1][0]=5;
conex[5][1][1]=0;

    //conexió Pr3
conex[6][0][0]=4;
conex[6][0][1]=2;
conex[6][1][0]=6;
conex[6][1][1]=0;

    //conexió Main
conex[7][0][0]=5;
conex[7][0][1]=2;
conex[7][1][0]=7;
conex[7][1][1]=1;
conex[7][2][0]=10;
conex[7][2][1]=0;

    //conexió2 Maout
conex[8][0][0]=6;
conex[8][0][1]=2;
conex[8][1][0]=8;
conex[8][1][1]=1;
conex[8][2][0]=11;
conex[8][2][1]=0;

    //conexió Tr1
conex[9][0][0]=7;
conex[9][0][1]=0;
conex[9][1][0]=9;
conex[9][1][1]=3;

```

```

//conexió Tr2
conex[10][0][0]=8;
conex[10][0][1]=0;
conex[10][1][0]=9;
conex[10][1][1]=0;

//inicializamos x[MAX][MAX][MAX]

i=0;
j=0;
z=0;
while(i<=MAX)
{
    j=0;
    while(j<3)
    {
        z=0;
        while(z<=MAX)
        {
            x[i][j][z]=MAX-1;

            z++;
        }
        j++;
    }
    i++;
}

//Creacion de modulos

Afterburner aft2;

//Se debe modificar al aumentar el número de ecuaciones
//Definición de los modulos a introducir
double dF[MAX][MAX], dF_copy[MAX][MAX];
double Fm[MAX][MAX]={Tin, Tout, fa, Pin},
{nc,fa,ft},
{mout,min,mfuel},
{nd,Pr1},
{Pr1,Pr2,Pr3},
{Pr2,gamma,Main},
{Pr3,gamma,Maout},
{Tr1,Main,gamma},
{Tr2, Maout, gamma},
{Tr2,Tout,Tin,Tr1},
{Main,vin,gamma,R,Tin},
{Maout,vout,gamma,R,Tout},
{fa,min,mfuel}

};

//Copiamos Fx en dfx
i=0;
j=0;
while(i<=MAX)
{
    j=0;
    while(j<=MAX)
    {
        dF[i][j]=Fm[i][j];
        dF_copy[i][j]=Fm[i][j];
    }
}

```

```

        j++;
    }
    i++;
}

//inicializamos sol,F;
i=0;
while(i<=MAX)
{
    sol[i]=0;
    F[i]=1;
    i++;
}

//Inicializamos links
i=0;
j=0;
while(i<MAX)
{
    j=0;
    while(j<MAX)
    {
        links[i][j]=MAX-1;

        j++;
    }
    i++;
}

//Inicializamos y definimos los diferenciales
i=0;
j=0;
c=0;
cont=0;

while(i<MAX)
{
    j=0;
    while(j<MAX)
    {
        if(Fm[i][j]==-1)
        {
            links[i][c]=j;
            Fm[i][j]=aft2.Afterburner2_guess(i,j);
            dF[i][j]=aft2.Afterburner2_guess(i,j);
            dF_copy[i][j]=aft2.Afterburner2_guess(i,j);
            c++;
            cont++;
        }
        j++;
    }
    c=0;
    i++;
}

n=cont-nc1;    //Número de incognitas reales

//Montamos la matriz dFx para localizar las incognitas a derivar
////////////////////////////////////
```

```

//FILTRO

r=0;
c=0;
cont=1;
xn=0;
i=0;
j=0;
z=0;
int contador=0,c1=0,c2=0,c3=0;
bool fsec=false,fpr=false;

while(i<MAX)
{
    j=0;
    while(j<MAX)
    {
        fpr=false;
        fsec=false;
        if(links[i][j]!=(MAX-1))
        {
            for(z=0;z<=nc1;z++)
            {
                if(((i==conex[z][1][0])&&(links[i][j]==conex[z][1][1]))||((i==conex[z][2][0])&
&(links[i][j]==conex[z][2][1])))
                    {c1++;

                        fsec=true;
                        //if(links[i][j+1]==MAX-1)    cont++;
                    }
            }
        }

        if((i==conex[cont][0][0])&&(links[i][j]==conex[cont][0][1])&&(fsec==false))
        {
            c2++;
            x[xn][r][c]=conex[cont][0][0];
            x[xn][r][c+1]=conex[cont][0][1];

            x[xn][r+1][c]=conex[cont][1][0];
            x[xn][r+1][c+1]=conex[cont][1][1];

            x[xn][r+2][c]=conex[cont][2][0];
            x[xn][r+2][c+1]=conex[cont][2][1];

            fpr=true;
            xn++;
            cont++;
            //if(links[i][j+1]==MAX-1)    cont++;
        }

        if((fsec==false)&&(fpr==false))
        {
            c3++;
            x[xn][r][c]=i;
            x[xn][r][c+1]=links[i][j];
            xn++;
            if(links[i][j]==MAX-1)    cont++;
        }
    }
}

```

```

        j++;
    }
    i++;
}

bool in1=false, in2=false, in3=false;
int jj=0;
int ii=0;
int cont1=0;
//Bucle que reevalúa el método de Newton y no sale de la condición hasta que
se alcanza la precisión impuesta
while((abs(F[0])>0.000001)||abs(F[1])>0.000001)||abs(F[2])>0.000001)||abs(F
[3])>0.000001)||abs(F[4])>0.000001)||abs(F[5])>0.000001)||abs(F[6])>0.000001)||abs(F
[7])>0.000001)||abs(F[8])>0.000001)||abs(F[9])>0.000001)||abs(F[10])>0.000001)||
abs(F[11])>0.000001)||abs(F[12])>0.000001))
{

    i=0;
    j=0;
    z=0;
    r=0;
    c=0;
    cont=0;
    cont1=0;
    while(i<(n*n))
    {

        if(in3==false)
        {
            while(cont<n)
            {
                while(r<3)
                {

                    dF[x[cont][r][c]][x[cont][r][c+1]]=aft2.Afterburner2_guess(x[cont][r][c],x[con
t][r][c+1])+ delta_x;

                                r++;
                }

                cont++;

                z=0;
                r=0;
                break;
            }
        }
        else
        {
            while(cont1<n)
            {
                Fm[x[cont1][0][0]][x[cont1][0][1]]=sol[cont1];

                Fm[x[cont1][1][0]][x[cont1][1][1]]=Fm[x[cont1][0][0]][x[cont1][0][1]];

                Fm[x[cont1][2][0]][x[cont1][2][1]]=Fm[x[cont1][0][0]][x[cont1][0][1]];

                dF_copy[x[cont1][0][0]][x[cont1][0][1]]=sol[cont1];

                dF_copy[x[cont1][1][0]][x[cont1][1][1]]=dF_copy[x[cont1][0][0]][x[cont1][0][1]
];

                dF_copy[x[cont1][2][0]][x[cont1][2][1]]=dF_copy[x[cont1][1][0]][x[cont1][1][1]
];

                dF[x[cont1][0][0]][x[cont1][0][1]]=sol[cont1]+delta_x;

```

```

dF[x[cont1][1][0]][x[cont1][1][1]]=dF[x[cont1][0][0]][x[cont1][0][1]];
dF[x[cont1][2][0]][x[cont1][2][1]]=dF[x[cont1][1][0]][x[cont1][1][1]];

        cont1++;
        break;
    }
    z=0;

}

//Se debe modificar al aumentar el número de ecuaciones
//Calculo del Jacobiano y almacenaje en dfx[]

    dfx[i]=(aft2.Afterburner2_f1(dF[z][0],dF[z][1],dF[z][2],dF[z][3])-
aft2.Afterburner2_f1(Fm[z][0],Fm[z][1],Fm[z][2],Fm[z][3]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f2(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f2(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f3(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f3(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f4(dF[z][0],dF[z][1])-
aft2.Afterburner2_f4(Fm[z][0],Fm[z][1]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f5(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f5(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f6(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f6(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f7(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f7(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f8(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f8(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f9(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f9(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;
    z++;

    dfx[i]=(aft2.Afterburner2_f10(dF[z][0],dF[z][1],dF[z][2],dF[z][3])-
aft2.Afterburner2_f10(Fm[z][0],Fm[z][1],Fm[z][2],Fm[z][3]))/delta_x;
    i++;
    z++;

```

```

    dfx[i]=(aft2.Afterburner2_f11(dF[z][0],dF[z][1],dF[z][2],dF[z][3],dF[z][4])-
aft2.Afterburner2_f11(Fm[z][0],Fm[z][1],Fm[z][2],Fm[z][3],Fm[z][4]))/delta_x;
    i++;
    z++;

```

```

    dfx[i]=(aft2.Afterburner2_f12(dF[z][0],dF[z][1],dF[z][2],dF[z][3],dF[z][4])-
aft2.Afterburner2_f12(Fm[z][0],Fm[z][1],Fm[z][2],Fm[z][3],Fm[z][4]))/delta_x;
    i++;
    z++;

```

```

    dfx[i]=(aft2.Afterburner2_f13(dF[z][0],dF[z][1],dF[z][2])-
aft2.Afterburner2_f13(Fm[z][0],Fm[z][1],Fm[z][2]))/delta_x;
    i++;

```

```

//Reinicializamos todo el sistema

```

```

a=0;

```

```

b=0;

```

```

int contx=0;

```

```

while(a<MAX)

```

```

{

```

```

    b=0;

```

```

    while(b<MAX)

```

```

    {

```

```

if(dF[a][b]==dF_copy[a][b]+delta_x)

```

```

    {

```

```

        contx++;

```

```

        dF[a][b]=dF_copy[a][b];

```

```

    }

```

```

    b++;

```

```

}

```

```

a++;

```

```

}

```

```

}

```

```

//Realimentacion del metodo de newton rapshon

```

```

//Para la primera iteración entrará en el primer if y copiará los
valores de Fm en sol

```

```

//En las siguientes iteraciones los valores de Fm se copiarán en sol y
así realimentaremos el sistema.

```

```

if(in2==false)

```

```

{

```

```

    cont1=0;

```

```

    while(cont1<=n)

```

```

    {

```

```

        sol[cont1]=Fm[x[cont1][0][0]][x[cont1][0][1]];

```

```

        cont1++;

```

```

    }

```

```

    in2=true;

```

```

}

```

```

else

```

```

{

```

```

    cont1=0;

```

```

    while(cont1<=n)

```

```

    {

```

```

        Fm[x[cont1][0][0]][x[cont1][0][1]]=sol[cont1];

```

```

        Fm[x[cont1][1][0]][x[cont1][1][1]]=Fm[x[cont1][0][0]][x[cont1][0][1]];

```



```

Fm[x[cont1][2][0]][x[cont1][2][1]]=Fm[x[cont1][0][0]][x[cont1][0][1]];
    cont1++;
}
}

//Se debe modificar al aumentar el número de ecuaciones

// f1
F[0]=aft2.Afterburner2_f1(Fm[0][0],Fm[0][1],Fm[0][2],Fm[0][3]);
// f2
F[1]=aft2.Afterburner2_f2(Fm[1][0],Fm[1][1], Fm[1][2]);
// f3
F[2]=aft2.Afterburner2_f3(Fm[2][0],Fm[2][1],Fm[2][2]);
// f4
F[3]=aft2.Afterburner2_f4(Fm[3][0],Fm[3][1]);
// f5
F[4]=aft2.Afterburner2_f5(Fm[4][0],Fm[4][1],Fm[4][2]);
// f6
F[5]=aft2.Afterburner2_f6(Fm[5][0],Fm[5][1],Fm[5][2]);
// f7
F[6]=aft2.Afterburner2_f7(Fm[6][0],Fm[6][1], Fm[6][2]);
// f8
F[7]=aft2.Afterburner2_f8(Fm[7][0],Fm[7][1],Fm[7][2]);
// f9
F[8]=aft2.Afterburner2_f9(Fm[8][0],Fm[8][1],Fm[8][2]);
// f10
F[9]=aft2.Afterburner2_f10(Fm[9][0],Fm[9][1],
Fm[9][2],Fm[9][3]);
// f11

F[10]=aft2.Afterburner2_f11(Fm[10][0],Fm[10][1],Fm[10][2],Fm[10][3],Fm[10][4])
;
// f12

F[11]=aft2.Afterburner2_f12(Fm[11][0],Fm[11][1],Fm[11][2],Fm[11][3],Fm[11][4])
;
// f13
F[12]=aft2.Afterburner2_f13(Fm[12][0],Fm[12][1],Fm[12][2]);

//Copia de matrices A -> dfx
int xx=0,yy=0;
while (xx<n*n)
{
    A[xx]=dfx[xx];
    xx++;
}

//Copia de matrices dFx -> F
while (yy<n)
{
    dFx[yy]=F[yy];
    yy++;
}

integer N = n;           //Numero de columnas de A
integer nrhs = 1;         //Numero de columnas de B
integer lda = n;          //Numero de filas de A
integer ipiv[20];
integer ldb = n;          //Numero de filas de B
integer info;

```

```

    dgesv_(&N, &nrhs, A, &lda, ipiv, dFx, &ldb, &info);

    if(info == 0) /* succeed */
    {
        printf("The solution is %lf %lf %lf %lf %lf %lf %lf %lf %lf\n",
F[0], F[1], F[2], F[3], F[4], F[5], F[6], F[7], F[8], F[9], F[10], F[11], F[12]);
    }
    else
        fprintf(stderr, "dgesv_ fails %d\n", info);

    ii=0;
    //Aplicacion directa de la ecuación del método de Newton para un
sistema de matrices.
    while(ii<n)
    {
        sol[ii]=sol[ii]-dFx[ii];
        ii++;
    }
    in3=true;
    jj++;

}

    printf("The final solution is %lf\n %lf\n %lf\n %lf\n %lf\n %lf\n %lf\n %lf\n
%lf\n %lf\n %lf\n %lf\n %lf\n", sol[0], sol[1], sol[2], sol[3], sol[4], sol[5],
sol[6], sol[7], sol[8] ,sol[9], sol[10], sol[11], sol[12]);
    printf("\nIterations %d\n",jj);

    system("pause");
    exit(0);

```